SEMI-MARKOV MODELS FOR EVALUATING NAVAL TACTICAL DECEPTION CONCEPTS

James M. Moore

Stanford Research Institute

Prepared for:

Office of Naval Research

April 1973

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D AD 778 511

Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author)
Stanford Research Institute
333 Ravenswood Avenue
Menlo Park, California 94025

2a. REPORT SECURITY CLASSIFICATION
UNCLASSIFIED

2b. GROUP

3 REPORT TITLE

SEMI-MARKOV MODELS FOR EVALUATING NAVAL TACTICAL DECEPTION CONCEPTS

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
Technical Note

5 AUTHOR(S) (First name, middle initial, last name)

James M. Moore

6 REPORT DATE
April 1973

7a. TOTAL NO. OF PAGES
127

7b. NO. OF REFS
10

8a. CONTRACT OR GRANT NO.
N00014-71-C-0419

b. PROJECT NO. RF018-02

c. NR 274-008-40

d.

9a. ORIGINATOR'S REPORT NUMBER(S)
NWRC TN-42

9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10 DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11 SUPPLEMENTARY NOTES

12. SPONSORING MILITARY ACTIVITY
Naval Analysis Programs (Code 431)
Office of Naval Research
Arlington, VA 22217

13 ABSTRACT

A semi-Markov model representation of a submarine searching for a high value target in a field of decoys is extended beyond previous work. The model has been used to assess the potentials of tactical deception concepts in modern antisubmarine warfare. The assessment results, including model sensitivity analyses, are published in a separate, classified, final project report. This technical note describes the details of the model's structure. The advantages of the earlier model are developed, and methods for numerical solution of the newer, larger model are examined. Model comparisons and sensitivity are explored. Suggestions for future extensions and modifications are provided.

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Search						
Detection						
FORMAC						
Markov						
Semi-Markov						
Decoy						
Submarines						
Antisubmarine						
Tactical Deception						

STANFORD RESEARCH INSTITUTE
Menlo Park, California 94025 · U.S.A.

Naval Warfare Research Center *April 1973*

Technical Note
NWRC-TN-42

SEMI-MARKOV MODELS FOR EVALUATING NAVAL TACTICAL DECEPTION CONCEPTS

By: JAMES M. MOORE

Prepared for:

NAVAL ANALYSIS PROGRAMS (CODE 431)
OFFICE OF NAVAL RESEARCH
ARLINGTON, VIRGINIA 22217

CONTRACT N00014-71-C-0419
Task NR274-008-40

SRI Project 1318-5

Approved by:

LAWRENCE J. LOW, *Director*
Naval Warfare Research Center

ABSTRACT

A semi-Markov model representation of a submarine searching for a high value target in a field of decoys is extended beyond previous work. The model has been used to assess the potentials of tactical deception concepts in modern antisubmarine warfare. The assessment results, including model sensitivity analyses, are published in a separate, classified, final project report. This technical note describes the details of the model's structure. The advantages of the earlier model are developed, and methods for numerical solution of the newer, larger model are examined. Model comparisons and sensitivity are explored. Suggestions for future extensions and modifications are provided.

PREFACE

The research reported on in this technical note was conducted as a subtask within a larger project directed toward the assessment of tactical deception techniques in surface force defense against cruise-missile armed submarines. The project was sponsored by Naval Analysis Programs, Mr. R. J. Miller, Director, in the Office of Naval Research. Mr. J. G. Smith was the ONR Project Scientific Officer.

The research effort was performed within the Undersea Warfare Program (Mr. A. Bien, Manager) of the Naval Warfare Research Center (Mr. L. J. Low, Director) of Stanford Research Institute. Mr. J. M. Moore succeeded Mr. M. W. Zumwalt as project leader.

The author wishes to acknowledge the assistance of Mr. A. Korsak in resolving some computational difficulties, and of Messrs S. Uralli and F. Rickey in implementing the FORMAC computer language.

SUMMARY

Introduction

Evaluation of tactical deception concepts in modern naval warfare is a fundamental concern of naval planners. This technical note presents the current status of an evolving family of analytical models designed to assist in these evaluations. This work supports a larger methodology assessment and concept evaluation presented in Reference 1. The reader should consult this classified reference for results using the models. At the present stage the models deal primarily with the search stage of encounter. However, their structure naturally leads to an expansion of the engagement aspects.

Problem Definition

The scenario for analysis involves a high value ship (HVS) operating within a certain specified area. Typically, the HVS is an aircraft carrier (CV) operating alone or with a carrier task group (CTG). Operating within the same area are a number of low value devices called decoys which have characteristics similar to the HVS or task group. A submarine enters the area at time zero and begins looking for the HVS in order to engage it. In the search process the searcher encounters the decoys. Because the destruction of a target will involve the expenditure of a limited resource, as well as the possible compromise of concealment, the submarine does not wish to destroy decoys; he must therefore spend time to classify a target as high value or decoy. The submarine wants to minimize search time, while the HVS wishes to maximize it. The targets are not able to observe the searcher.

The following questions about this situation are of concern:

(1) What risk of being found does the HVS incur over a given period of time?

(2) What are the relative effects on risk of decoy characteristics, such as number, detectability, mobility, and fidelity?

These notions involve dynamic probabilistic systems and they are discussed in terms of semi-Markov random processes.

Approach

The problem of search in the presence of decoys naturally lends itself to representation by a semi-Markov process because the state of the searcher and the transitions between states can be well defined. Placing the problem in the format of a more general semi-Markov process provides a framework which can accept hypothetical or experimental data that might not be well adapted to the more specialized Markov chain. A feature of the approach developed for this study is that the solutions to the various models considered are in closed form, amenable to straightforward calculation. The approach also provides a framework in which efficient event-step simulation models can be developed. The assumptions governing the structure of the models are presented in the following pages.

ASSUMPTIONS:

Geometry and Kinematics

- There is a single submarine opposing several HVSs and decoys; typically, there is one HVS, which is a CV operating alone or in a CTG
- All activity takes place in a circular operating area of known fixed size
- The submarine is initially in the operating area with no contacts at time zero
- At any fixed instant in time, positions of HVSs and decoys are distributed over the operating area according to a uniform probability distribution
- Over short intervals of time, everything moves in straight lines with uniformly random headings
- Stochastic independence of motion is assumed among decoys as a group, among HVSs as a group, and between the two groups, except in the case of reduced overlap when HVSs avoid decoys on a probabilistic basis

ASSUMPTIONS:

Target Detection and Selection

- Submarine detection of an HVS, CTG, CTG element or decoy is represented by a definite range law
- The centers of all detection circles lie inside the operating area (i.e., the searcher as well as the HVS knows what the operating area is)
- When presented with an array of targets (both decoys and HVSs) from which one is to be selected for classification, the targets are all equally likely to be chosen (i.e., the decoys are identical and are indistinguishable from the identical HVSs until a classification is made)
- There are no acoustic masking effects

ASSUMPTIONS:

Target Classification

- Misclassification--a final declaration of the target to be other than it really is--occurs with a probability that may depend on true target type
- Once a target is classified as a decoy, that target is removed from the target field as long as information on it is retained
- Information on a classified target is retained until the end of a random length of time or until another target is detected, whichever occurs first
- Once a target is selected, the submarine stays with that one target until classification is made
- The classification process is represented by a random length of time immediately preceding a final decision on target type
- The final decision on target type is dichotomous--HVS or decoy

ASSUMPTIONS:

Engagement

- If the engagement phase is considered, it is assumed the submarine allows itself only one attack in which all its weapons are launched; hence, an attack on a misclassified decoy removes the submarine as a threat to the HVS
- The game ends when a target (HVS or decoy) is classified as HVS (and the submarine launches its weapons) or time runs out

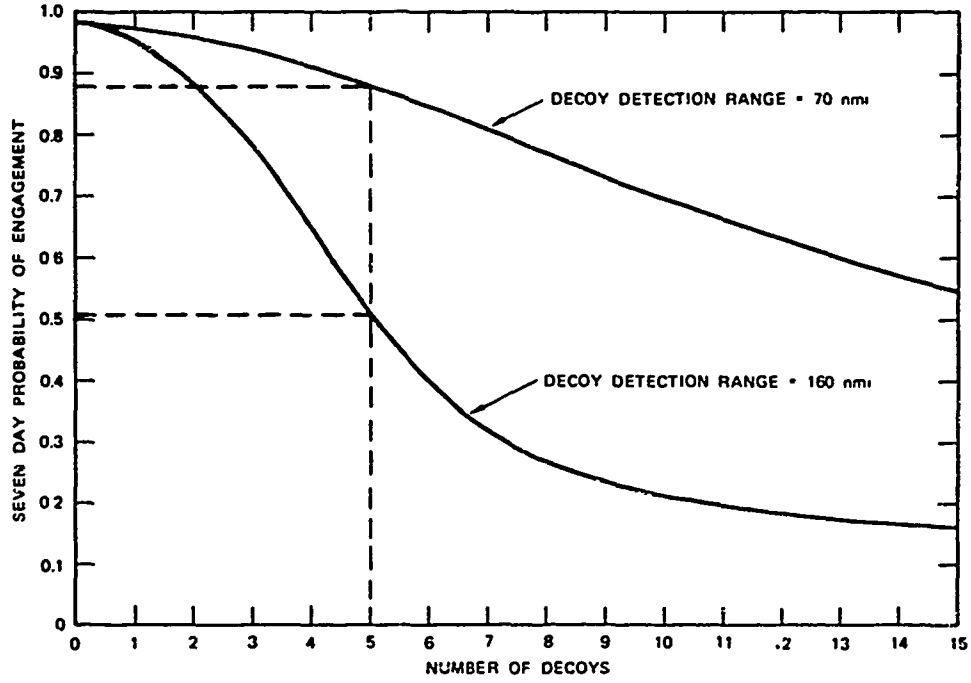
Exponential holding times in search states are assumed throughout. Holding times in classification states may have more general distribution. In computer implementation there are actually two models--one has misclassification capability, the other has the more general capture time distribution capability. However, it is often convenient to consider the two as one, with the above two capabilities mutually exclusive.

Simulation studies, conducted as an adjunct to both the original three-state model formulation and the current study, have demonstrated that the assumed exponential search time distribution and the adopted model for determining mean time to target detection are in fact valid with respect to the assumptions made concerning the search process (primarily the definite range detection law and random motion).

Two computational approaches are considered. One is based on partial fraction expansion of exponential transforms of probability functions. This approach involved an experimental effort to use the IBM computer language FORMAC (p.59) to execute the algebraic manipulations involved. Partial fraction expansion allows consideration of non-exponential holding times. When all holding times are assumed exponential, a more efficient approach based on a Pade' approximation for a matrix exponential is utilized. There are two major reasons to consider the partial fraction expansion: (1) to study sensitivity to distributional form, especially variance, and (2) to study characteristics other than probability of state as a function, such as mean time in state and transition counting.

Sample Results

An example result addressing the questions posed in the problem definition is shown in the figure below. The situation represented is a 15 knot CTG in an acoustic convergence zone; the decoys are non-moving



PROBABILITY OF ENGAGEMENT AS A FUNCTION OF NUMBER OF DECOYS

and capture the submarine for an average of 10 hours per decoy/submarine encounter. Probability of a submarine/HVS engagement over a seven day period is displayed as a function of the number of decoys for two different decoy detection ranges. With no decoys deployed the HVS has probability one of an encounter with the submarine. If five decoys are deployed, this probability falls to 0.88 if it is a small device and 0.51 if it is a large device.

Conclusions and Recommendations

The models discussed in this document are implemented as operational computer programs and have been used in concept evaluation studies reported in Reference 1. A comprehensive sensitivity analysis of the models is presented in that reference. Experience in the development and use of these models allows some general conclusions to be drawn about the methodology.

Two conclusions about the significance of problem components are easy to draw: (1) potential for target misclassification is a highly significant part of the problem, and (2) retention of information on the last target classified is insignificant. Also very significant is what has been discovered about the importance of geometry. Geometry representation can have an appreciable effect, but the effect is due primarily to the representation of the dissipation of swept area. The following simplified approach is suggested for future work:

(1) Ignore boundary conditions

(2) Adopt simple modification of unbounded area detection rates via approximations based on Figure 3.10 or an extension of this figure.

A conclusion about method of analysis is that the FORMAC computer language is an interesting and powerful tool for analyzing continuous time semi-Markov process, but it can become cumbersome and time consuming in itself. Its use is best restricted to a small number of states (certainly no more than six), or to cases of very sparse transition matrices, where it can be very useful in deriving compact closed form solutions. However, for a larger number of states it appears a discrete time approximation is the better approach where consideration of non-Markovian processes is desirable. In the work described here it seems a 2-hour discrete time step would be the minimum necessary and perhaps longer, say 5 hours, would be acceptable.

CONTENTS

pages vi, vii and viii Blank

ILLUSTRATIONS

TABLES

1. INTRODUCTION

Evaluation of tactical deception concepts in modern naval warfare is a fundamental concern of naval planners. This technical note presents the current status of an evolving family of analytical models designed to assist in these evaluations. At the present stage the models deal primarily with the search stage of encounter. However, their structure naturally leads to an expansion of the engagement aspects.

The scenario for analysis involves a high value ship (HVS) operating within a certain specified area. Typically, the HVS is an aircraft carrier (CV) operating alone or with a carrier task group (CTG). Operating within the same area are a number of low value devices called decoys which have characteristics similar to the HVS or task group. A submarine enters the area at time zero and begins looking for the HVS in order to engage it. In the search process the searcher encounters the decoys. Because the destruction of a target will involve the expenditure of a limited resource, as well as the possible compromise of concealment, the submarine does not wish to destroy decoys; he must therefore spend time to classify a target as high value or decoy. The submarine wants to minimize search time, while the HVS wishes to maximize it. The targets are not able to observe the searcher. The following questions about this situation are of concern:

(1) What risk of being found does the HVS incur over a given period of time?

(2) What are the relative effects on risk of decoy characteristics, such as number, detectability, mobility, and fidelity?

These notions involve dynamic probabilistic systems. They will be discussed in terms of semi-Markov random processes.

The models presented here have been developed as a subtask on a larger project that is directed toward evaluating deception concepts in surface force defense against cruise-missile armed submarines. The principal document of the larger project is the final report (Ref. 1),* entitled:

> "Naval Tactical Deception Concepts: Methodology Assessment and Concept Evaluation (U)."

Reference 2, entitled:

> "Simulation Model for Evaluation of Naval Tactical Deception Concepts (U),"

and this technical note are supporting documents for the final report. The work reported in this document is a continuation of earlier work reported in Ref. 3, entitled:

> "Semi-Markov Models of Search in the Presence of Decoys."

Reference 3 concluded that the semi-Markov process is an effective framework in which to study questions regarding the capabilities of decoys to delay a searcher looking for a particular type of target.

The problem of search in the presence of decoys naturally lends itself to representation by a semi-Markov process because the state of the searcher and the transitions between states can be well defined. Placing the problem in the format of a more general semi-Markov process provides a framework which can accept hypothetical or experimental data that might not be well adapted to the more specialized Markov chain.

---

*References are listed at the end of this report.

A feature of the semi-Markov approach developed for this study is that the solutions to the various models considered are in closed form, amenable to straightforward calculation. The approach also provides a framework in which efficient event-step simulation models can be developed. A discussion of general Markov and semi-Markov processes is available in Ref. 4.

The previous work evolved a three-state model and developed an analysis of the value of acoustic countermeasures in surface force defense (Ref. 5). This three-state model is able to provide some insight into the problem, but it has some shortcomings, specifically:

- Detection radii are assumed to be small in comparison to the size of the operating area
- There is a requirement that either HVS speed or decoy capture time,* or both, be large enough to dissipate the area swept by the submarine prior to encountering a decoy
- A classified decoy immediately returns to the target field after classification, i.e., the submarine does not retain any information from a classification
- Perfectly reliable classification is assumed
- The implemented model assumed exponentially distributed classification times, even though the methodology of semi-Markov processes provides the mechanism for more generality.

The modeling effort reported on here has concentrated on deriving and implementing solutions to these problem areas. The following assumptions govern the structure of the new model:

---

*Time spent by the submarine in classifying a decoy.

- There is a single submarine opposing several HVSs and decoys; typically, there is one HVS, which is a CV operating alone or in a CTG
- All activity takes place in a circular operating area of known fixed size
- The submarine is initially in the operating area with no contacts at time zero
- At any fixed instant in time, positions of HVSs and decoys are distributed over the operating area according to a uniform probability distribution
- Over short intervals of time, everything moves in straight lines with uniformly random headings
- Stochastic independence of motion is assumed among decoys as a group, among HVSs as a group, and between the two groups, except in the case of reduced overlap when HVSs avoid decoys on a probabilistic basis
- Submarine detection of an HVS, CTG, CTG element or decoy is represented by a definite range law
- The centers of all detection circles lie inside the operating area (i.e., the searcher as well as the HVS knows what the operating area is)
- When presented with an array of targets (both decoys and HVSs) from which one is to be selected for classification, the targets are all equally likely to be chosen (i.e., the decoys are identical and are indistinguishable from the identical HVSs until a classification is made)
- There are no acoustic masking effects
- Once a target is selected, the submarine stays with that one target until a classification is made
- The classification process is represented by a random length of time immediately preceding a final decision on target type
- The final decision on target type is dichotomous--HVS or decoy
- If the engagement phase is considered, it is assumed the submarine allows itself only one attack in which all its weapons are launched; hence, an attack on a misclassified decoy removes the submarine as a threat to the HVS

- Misclassification--a final declaration of the target to be other than it really is--occurs with a probability that may depend on true target type
- Once a target is classified as a decoy, that target is removed from the target field as long as information on it is retained
- Information on a classified target is retained until the end of a random length of time or until another target is detected, whichever occurs first
- The game ends when a target (HVS or decoy) is classified as HVS (and the submarine launches its weapons) or time runs out.

Exponential holding times in search states are assumed throughout. Holding times in classification states may have more general distribution. In computer implementation there are actually two models--one has misclassification capability, the other has the more general capture time distribution capability. However, it is often convenient to consider the two as one, with the above two capabilities mutually exclusive.

Simulation studies, conducted as an adjunct to both the original three-state model formulation and the current study, have demonstrated that the assumed exponential search time distribution and the adopted model for determining mean time to target detection are in fact valid with respect to the assumptions made concerning the search process (primarily the definite range detection law and random motion).

2. MODEL STRUCTURE

The search problem is considered in terms of a dynamic system described by its state variable. The system is the submarine, and the state variable of interest is the occupation of its sensor/information-processing apparatus. Two mutually exclusive occupations are basic to the analysis: search and classification. The further partitioning of these states, and the transitions between states is the subject of the following paragraphs. First, the general semi-Markov process is reviewed, and then the models are developed.

2.1 The Continuous Time Semi-Markov Process

The semi-Markov process (SMP) is a stochastic process in which time is the independent variable, and the dependent variable can assume only a denumerable number of discrete values. A given value of the dependent variable is a state of the system, and the collection of all possible values is the state space. Change from a present state to a future state depends only on the present state and is governed by the transition probabilities of a Markov chain, while the time required for such a transition may, in general, depend on both the present state and the future state. Consider a space of functions $S = \{S_\omega(t);\ t \geq 0,\ \omega \in \Omega\}$. This space is the sample space and consists of functions of time, with the generic functional form parameterized by $\omega \in \Omega$. Thus, a sample "point" consists of the set of points $\{S_\omega(t);\ t \geq 0\}$; such a sample point is called a "realization" of the process and is denoted simply $S_\omega(\cdot)$. We will consider only a finite number of states, with a state denoted by one of the numbers 1, 2, ..., N. Thus, for a fixed value of t, $S_\omega(t)$ is in the set $\{1, 2, \ldots, N\}$ and specifies the state of the system at

that time. Hence, it is characteristic of the process that each realization $S_{\omega}(\cdot)$ is a staircase function; we will assume each realization continuous from the right. A typical realization is plotted in Figure 2.1 for a four-state SMP. The dots in the figure represent "events," that is, points where a transition into a state occurs. Transitions from a state into the same state may occur. If a transition from state i to state j ($i \neq j$) occurs at time τ, then $S_{\omega}(\cdot)$ will be continuous from the right but discontinuous from the left at $t = \tau$. Formally, we say the process is in state j at time t if the last event to occur was of type j, where $j \in \{1, \ldots, N\}$.

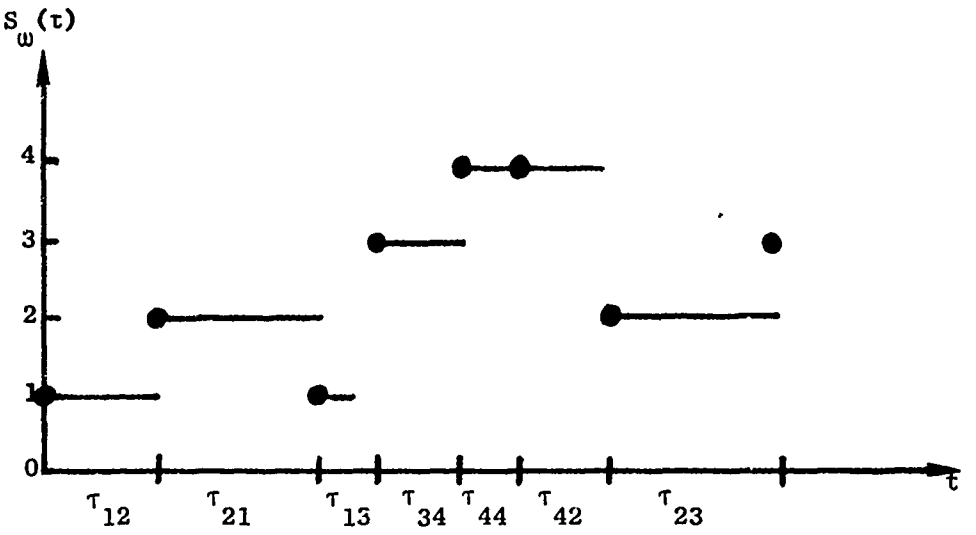


FIGURE 2.1 A TYPICAL REALIZATION OF A SMP

A SMP may be described in terms of the transition probability matrix $P = [p_{ij}]$ of its embedded Markov chain and a matrix of conditional holding time probability density functions $H(t) = [h_{ij}(t)]$. Whenever the process enters state i, the next state, j, to which it will move is determined according to state i's transition probabilities $p_{i1}, \ldots, p_{iN}$. After j has been determined, the process holds for a time τ_{ij} in state i.

The holding times τ_{ij} are positive valued random variables, each governed by a probability density function (pdf) $h_{ij}(\cdot)$.

It is convenient to use graphical concepts in constructing and analyzing models of semi-Markov processes. Two of these concepts are of particular interest here. One is the transition diagram, which is composed of a set of nodes numbered 1 to N representing the states and a set of directional arcs linking node i with node j for each positive p_{ij}. An example is shown in Figure 2.2 for a three-state process with a trapping state (a state from which there are no transitions other than to itself). Transition diagrams are useful in developing and describing the structure of a model. The other graphical concept of interest is the flow graph. The flow graph is composed of the same nodes and arcs as the transition possibility graph, but in addition each arc is labeled. The arc from node i to node j is labeled with $p_{ij}h^e_{ij}(s)$, where $h^e_{ij}(s)$ is the exponential transform of $h_{ij}(t)$, i.e.,

$$h^e_{ij}(s) = \int_0^\infty dt\, e^{-st} h_{ij}(t) \quad .$$

An example flow graph is shown in Figure 2.3. Flow graph concepts are useful in analysis of a structured model.

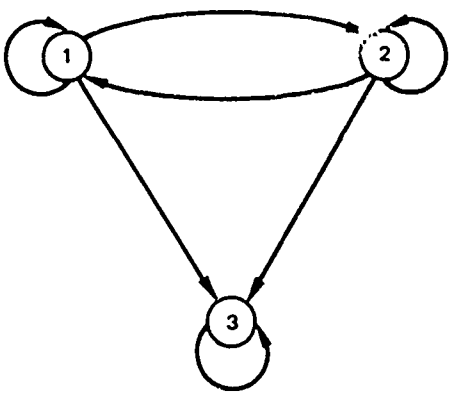


FIGURE 2.2 TRANSITION DIAGRAM

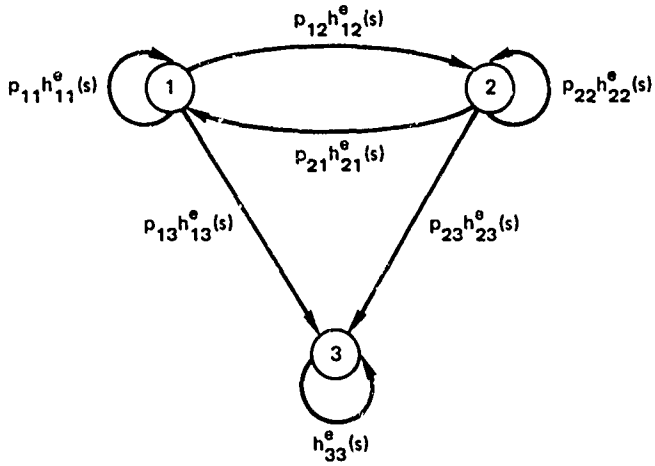


FIGURE 2.3 PROCESS FLOW GRAPH

Given the components P and H(t), or the process flow graph, we can define the interval transition probabilities $\varphi_{ij}(t)$. This quantity $\varphi_{ij}(t)$ is the conditional probability that the process will occupy state j at time t, given that it entered state i at time zero. Howard (Ref. 4) shows that

$$\varphi_{ij}(t) = \delta_{ij} {}^{>}w_i(t) + \sum_{k=1}^{N} p_{ik} \int_0^t d\tau \; h_{ik}(\tau)\varphi_{kj}(t-\tau)$$

$$\text{for } i,j = 1,\ldots,N \quad ; \quad t \geq 0 \quad .$$

where

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

$${}^{>}w_i(t) = 1 - \sum_{j=1}^{N} p_{ij} \int_0^t h_{ij}(\tau)d\tau \quad .$$

Other ways of obtaining $\varphi_{ij}(t)$ utilizing transforms, flow graph transmissions, and special assumptions on $h_{ij}(t)$ will be discussed in a later section.

In the modeling discussed in this document, only the class of transient semi-Markov processes are of interest. A transient process contains one or more trapping states. If state N is a trapping state, then the interval transition probability $\varphi_{iN}(t)$ is also the probability the process becomes trapped in state N at or before time t, given that it started in state i at time zero. In this interpretation as a trapping probability, $\varphi_{ij}(t)$ will be our measure of effectiveness.

2.2 The Six-State Model

The basic structure of the six-state model is shown in the transition diagram of Figure 2.4. The states are:

State 1. Entry: represents the initial search by the searcher prior to first encounter

State 2. Classify decoy (transient): represents capture of the submarine by a decoy subsequent to a search

State 3. Classify HVS: self-explanatory

State 4. Search (reduced field): represents search subsequent to classification of a decoy before information retained by the submarine on that decoy has dissipated

State 5. Search (full field): represents search with no information on decoys

State 6. Classify decoy (steady state): represents capture by a decoy immediately subsequent to capture by another decoy.

The entry state (State 1) provides a framework for analysis of searcher entry into the operating area. However, this particular aspect of the analysis has not been carried beyond the initial exploratory phase,

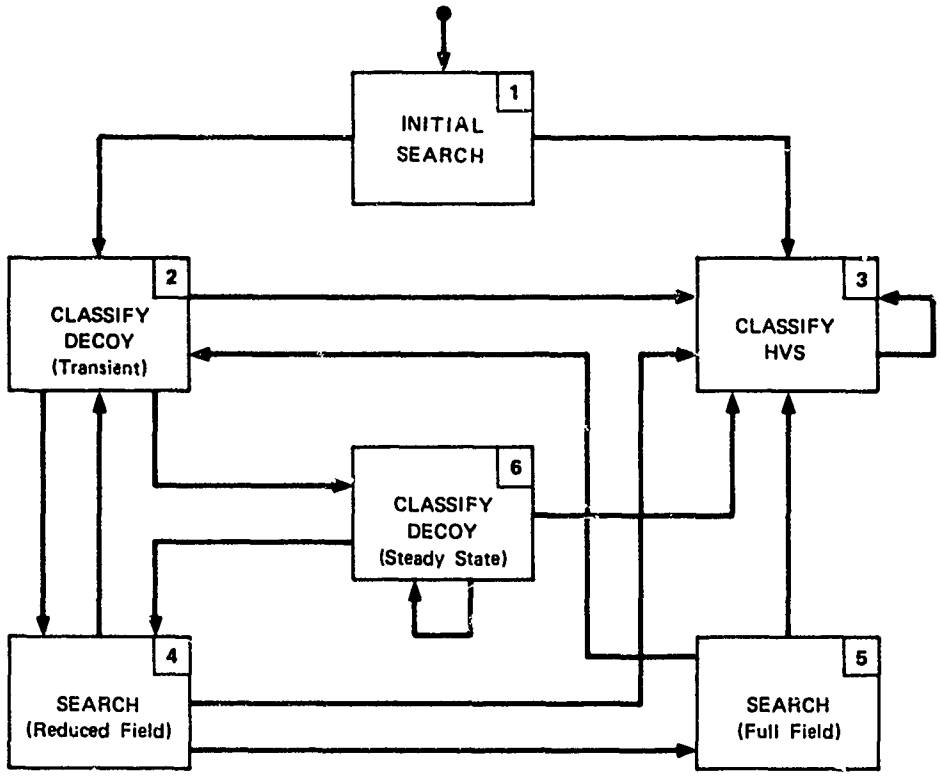


FIGURE 2.4 TRANSITION DIAGRAM FOR SIX STATE MODEL

where it was found the approach being developed would be very expensive computationally and have relatively small effect. The approach examined is discussed briefly in Appendix A. The result is that the searcher is assumed simply to be in the area initially without a target. The entry state is thus equivalent* to the search-full-field state (State 5). The derivation of the holding time densities and transition probabilities for these two states proceeds as follows. Define the random variables

*The size of the state space could easily be reduced from 6 to 5 by eliminating the redundancy of having two equivalent states. The reduction is not made here because this technical note is documentation for computer programs which retain the two equivalent states as a matter of convenience.

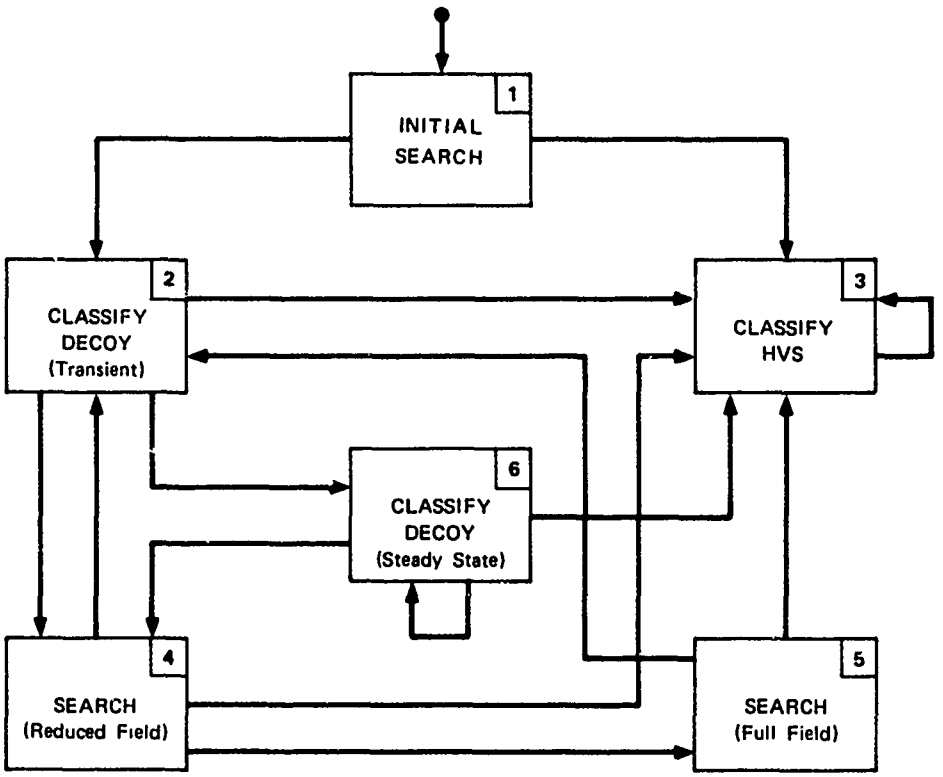


FIGURE 2.4 TRANSITION DIAGRAM FOR SIX STATE MODEL

where it was found the approach being developed would be very expensive computationally and have relatively small effect. The approach examined is discussed briefly in Appendix A. The result is that the searcher is assumed simply to be in the area initially without a target. The entry state is thus equivalent* to the search-full-field state (State 5). The derivation of the holding time densities and transition probabilities for these two states proceeds as follows. Define the random variables

*The size of the state space could easily be reduced from 6 to 5 by eliminating the redundancy of having two equivalent states. The reduction is not made here because this technical note is documentation for computer programs which retain the two equivalent states as a matter of convenience.

T_D = time to detect a decoy

T_H = time to detect a HVS,

where the pdf of T_H is $f_H(\cdot)$ and the pdf of T_D is $f_D(\cdot)$. We are now in the context of what Howard (Ref. 4) calls the competing process model. The pdf's $f_H(\cdot)$ and $f_D(\cdot)$ are the pdf's of the competing random variables. Using Howard's result and assuming

$$f_H(t) = \beta_H e^{-\beta_H t} \quad , \quad t \geq 0 \quad ,$$

$$f_D(t) = \beta_D e^{-\beta_D t} \quad , \quad t \geq 0 \quad ,$$

it is easy to show that

$$p_{12} = \frac{\beta_D}{\beta_D + \beta_H} \quad \text{and} \quad p_{13} = \frac{\beta_H}{\beta_D + \beta_H}$$

and that

$$h_{12}(t) = h_{13}(t) = (\beta_D + \beta_H)e^{-(\beta_D + \beta_H)t} \quad , \quad t \geq 0 \quad .$$

The interpretation of β_H and β_D is that they are rates at which the HVS and decoys, respectively, are encountered. The derivation of β_H and β_D will be discussed subsequently in Section 3.3. Suffice it to say here that $\beta_D = n_D\lambda_D$, where λ_D is the rate when there is one decoy, and n_D is the number of decoys; and $\beta_H = n_H\lambda_H$, where λ_H is the rate when there is one HVS and n_H is the number of HVSs

$$p_{12} = \frac{n_D\lambda_D}{n_D\lambda_D + n_H\lambda_H} \quad \text{and} \quad p_{13} = \frac{n_H\lambda_H}{n_D\lambda_D + n_H\lambda_H}$$

and

$$h_{12}(t) = h_{13}(t) = (n_D\lambda_D + n_H\lambda_H)e^{-(n_D\lambda_D + n_H\lambda_H)} , \quad t \geq 0$$

Since State 5 is considered equivalent to State 1 we have

$$p_{52} = p_{12} \quad \text{and} \quad p_{53} = p_{13}$$

and also

$$h_{52}(\cdot) = h_{53}(\cdot) = h_{12}(\cdot) = h_{13}(\cdot) \quad .$$

The search-reduced-field state (State 4) is similar to the search-full-field state (State 5) except the number of decoys is reduced by one and an information retention time T_R with pdf $f_R(\cdot)$ is included. Assuming

$$f_R(t) = \lambda_R e^{-\lambda_R t} , \quad t \geq 0$$

we can again apply the competing process concept. Here λ_R is an input and interpreted in the form of λ_R^{-1} as the mean information retention time. We obtain

$$p_{42} = \frac{(n_D-1)\lambda_D}{(n_D-1)\lambda_D + n_H\lambda_H + \lambda_R}$$

$$p_{43} = \frac{n_H\lambda_H}{(n_D-1)\lambda_D + n_H\lambda_H + \lambda_R}$$

$$p_{45} = \frac{\lambda_R}{(n_D-1)\lambda_D + n_H\lambda_H + \lambda_R}$$

and

$$h_{4i}(t) = [(n_D-1)\lambda_D + n_H\lambda_H + \lambda_R]e^{-[(n_D-1)\lambda_D + n_H\lambda_H + \lambda_R]t} , \quad t \geq 0$$

for $n_D \geq 1$ and i = 2, 3, 5. With $n_D = 0$ State 4 cannot be reached, so transition probabilities and holding times are not relevant.

The "classify decoy" states (2 and 6) are complicated in terms of transition probabilities. The two states differ in the geometrical considerations involved in specifying their transition probabilities. State 6 is designated as "steady state" because it assumes the dissipation of the area swept by the submarine (hence clear of decoys and HVS) prior to encountering a sequence of decoys. Figures 2.5 and 2.6 illustrate the situation. Assume for convenience of discussion that decoy and HVS have the same speed and detection range. In Figure 2.5 the submarine has encountered a decoy after a period of searching with no contacts. At the moment of encounter there is a "swept" circle (shaded),

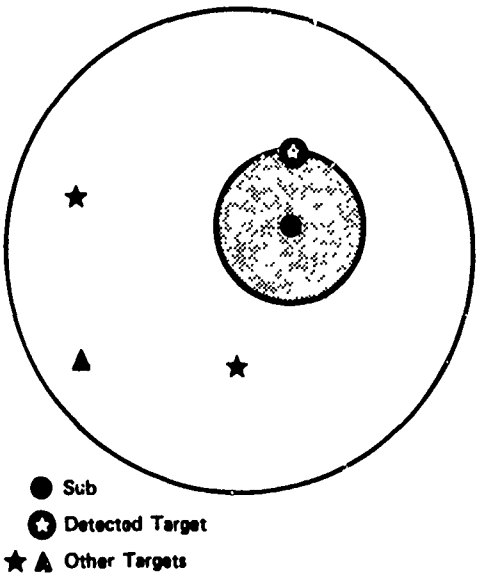


FIGURE 2.5 GEOMETRY AT TIME OF DETECTION

centered at the position of the submarine, which is free of targets except for the one just encountered on the boundary. That is, the probability of a target in the shaded circle is zero. Actually, the circle is an approximation of the true swept area, which trails behind the

submarine like a giant teardrop. The submarine does not know the position of targets outside this circle. Suppose the classification takes time t_o. Consider a hypothetical target just on the boundary of the shaded circle (see Figure 2.6). Moving at speed v_H toward the center for a time t_o, the hypothetical target reduces the radius of the shaded area by a distance $v_H t_o$.

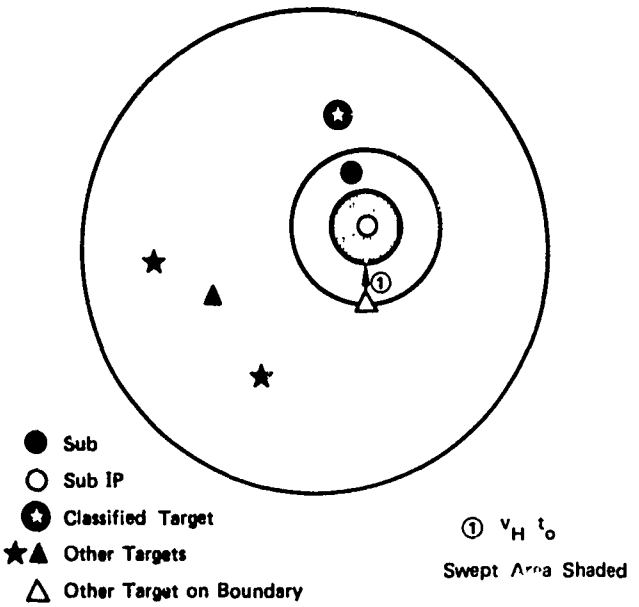


FIGURE 2.6 GEOMETRY AT TIME OF CLASSIFICATION

Hence, if $v_H t_o$ is small enough, when the submarine is through classifying the decoy there will be an area (the reduced shaded circle) where there is zero probability of targets existing. This is important for the following reason. We need the instantaneous probabilities P_D and P_H where

P_D = Pr {any given decoy is present}

P_H = Pr {any given HVS is present},

where "present" means within detection range of the submarine. The basic idea of the probabilities as obtained for State 2 is

$$P_D \approx \frac{A_D}{A_o - A_1}$$

$$P_H \approx \frac{A_H}{A_o - A_2} \quad ,$$

where

A_o = operating area

A_D = area enclosed by the intersection of A_o and the decoy detection circle

A_H = area enclosed by the intersection of A_o and the HVS detection circle

A_1 = area of reduced shaded circle for decoy

A_2 = area of reduced shaded circle for HVS.

In State 4 it is assumed that the swept (shaded) areas are dissipated completely, and hence $A_1 = A_2 = 0$. The inclusion of this swept area in State 2 and its dissipation in State 6 has found to be of high significance in some important cases. The actual definitions of P_D and P_H are more complicated than those given here and are discussed in detail in Section 3.1, along with the detailed derivations of the transition probabilities p_{2i} and p_{6i}. These details are postponed so that overall model structure can be seen more clearly.

The holding time pdf's for States 2 and 4 are arbitrarily assumed to be of the form

$$h_{2i}(t) = h_{6i}(t) = \frac{(\mu_D n)^n}{(n-1)!} t^{n-1} e^{-n\mu_D t} \quad , \quad t \geq 0 \quad (i = 3,4,6)$$

where $\mu_D \geq 0$ and n, a positive integer, are parameters of the distribution. This form is known as the Erlang distribution and its characteristics are discussed in Appendix B. Another way of stating this assumption is to say the capture time (call it T_{CD}) of a decoy has the specified form. Capture time of a target is the time spent by the submarine in classifying that target as a decoy or HVS. The assumption about T_{CD} is made for two reasons: (1) at the time of model construction no satisfactory model of the classification process was available, and (2) this form is particularly amenable to analysis and obtains a variety of shapes via proper parameter selection. Figure B.1 in Appendix B illustrates the range of shapes that can be achieved by changing n for a fixed value of μ_D. The mean of the distribution is μ_D^{-1} irrespective of the value of n. Hence, μ_D^{-1} is the mean capture time for a decoy.

State 3, "classify HVS," is the final state to be explained. This is the trapping state in the six-state model. The only nonzero transition probability is $p_{33} = 1$. Because the analytic measure we are using is basically time of first arrival in this state, it turns out that the form of the holding time pdf $h_{33}(\cdot)$ is irrelevant.

2.3 The Ten-State Model

The basic structure of the ten-state model is shown in the transition diagram of Figure 2.7. The basic difference between this and the six-state model is the inclusion here of the possibility of the submarine's misclassifying a target. This possibility generates a requirement for more states. The states are:

State 1. Entry: represents the initial search by the searcher prior to first encounter

State 2. Classify decoy (transient): represents capture of the submarine by decoy subsequent to search

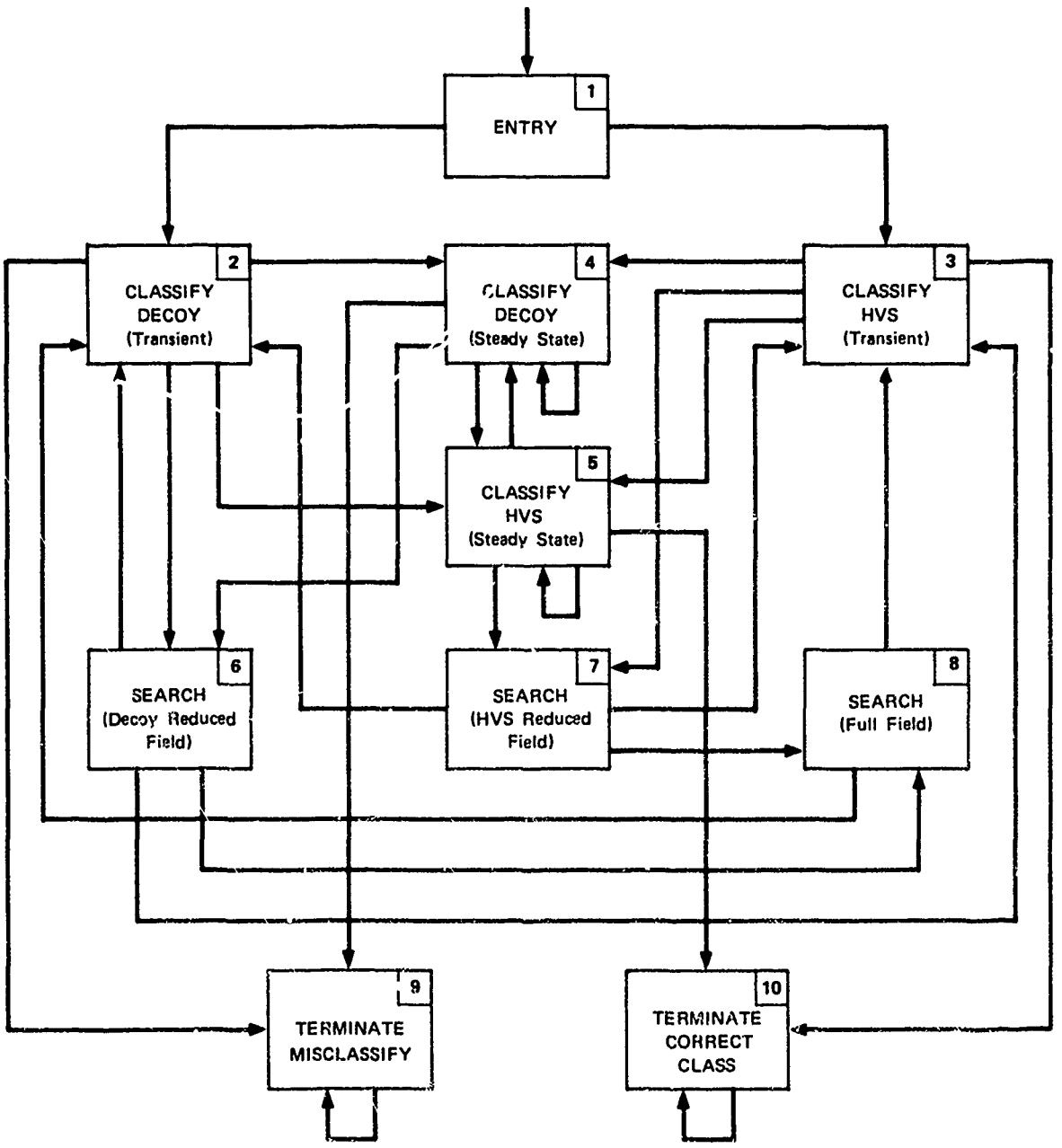


FIGURE 2.7 TRANSITION DIAGRAM FOR TEN STATE MODEL

State 3. Classify HVS: represents classification of an HVS subsequent to search

State 4. Classify decoy (steady state): represents capture of the submarine by a decoy immediately subsequent to correct classification of another decoy or misclassification of an HVS

State 5. Classify HVS (steady state): represents classification activity against an HVS immediately subsequent to the correct classification of a decoy or misclassification of another HVS

State 6. Search (decoy reduced field): represents search subsequent to (correct) classification of a decoy before information retained by the submarine on that decoy has dissipated (the decoy field is reduced by one)

State 7. Search (HVS reduced field): represents search subsequent to misclassification of an HVS before information retained by the submarine on that HVS has dissipated

State 8. Search (full field): represents search with no information on decoys

State 9. Terminate (misclassification): the game ends with a decoy incorrectly classified as an HVS

State 10. Terminate (correct classification): the game ends with an HVS correctly classified.

Note that the sequence of events modeled ends when the submarine classifies something as an HVS. This situation leaves something to be desired from the point of view of interpretation since subsequent action may be relevant. Two approaches are available. It can be considered that the framework is available on which to build models of action subsequent to classification. Or it can be assumed the submarine launches all of its missiles upon classification of a target as an HVS. In the latter case either the HVS is attacked or the submarine ceases to be a threat.

Most of the comments made on the six-state model apply directly here also. State 1 (entry) and its relations to State 2 and 3 are identical to the other model. Specifically,

$$p_{12}^* = \frac{n_D\lambda_D}{n_D\lambda_D + n_H\lambda_H} \quad \text{and} \quad p_{13}^* = \frac{n_H\lambda_H}{n_D\lambda + n_H\lambda_H}$$

and

$$h_{12}^*(t) = h_{13}^*(t) = (n_D\lambda_D + n_H\lambda_H)e^{-(n_D\lambda_D + n_H\lambda_H)t}, \quad t \geq 0 ,$$

where the superscript star (*) is used to distinguish ten-state entities from their six-state counterparts. The quantities λ_D and λ_H are as discussed for the six-state model. State 2, "classify decoy (transient)," and State 4, "classify decoy (steady state)," are very similar to States 2 and 6 of the six-state model. In particular, the holding time pdf's are

$$h_{2i}^*(t) = h_{4i}^*(t) = \mu_D e^{-\mu_D t}, \quad t \geq 0 \qquad (i = 4,5,6,9)$$

which is the form of $h_{2i}(\cdot)$ and $h_{6i}(\cdot)$ with n = 1. The geometry in the ten-state model is identical to that of the six-state model, so the transition probabilities p_{2i}^* and p_{4i}^* are adapted from p_{2i} and p_{6i}, with the required modification to allow for misclassification. Let p_{MD} be the probability the submarine misclassifies a decoy it investigates. Then

$$p_{2i}^* = p_{2i}(1 - p_{MD}) \quad \text{and} \quad p_{4i}^* = p_{6i}(1 - p_{MD}) \qquad (i = 4,5,6) ,$$

and

$$p_{29}^* = p_{49}^* = p_{MD} .$$

State 3, "classify HVS (transient)," and State 5, "classify HVS (steady state)," are conceptually equivalent to States 2 and 4, respectively. A different mean classification time $T_{CH} = \mu_H^{-1}$ is allowed for HVSs so that

$$h^*_{3i}(t) = h^*_{5i}(t) = \mu_H e^{-\mu_H t}, \quad t \geq 0 \qquad (i = 4,5,7,10) \quad .$$

Letting p_{MH} be the probability of misclassifying a HVS, we obtain p^*_{3i} and p^*_{5i} in a way that is strictly analogous to obtaining p^*_{2i} and p^*_{4i}. Quantities p'_{3i} and p'_{5i} are computed in a manner similar to that for p_{2i} and p_{6i}, with exceptions noted in Section 3.2. Then

$$p^*_{3i} = p'_{3i}(1 - p_{MH}) \quad \text{and} \quad p^*_{5i} = p'_{5i}(1 - p_{MH}) \qquad (i = 4,5,7)$$

and

$$p^*_{310} = p^*_{410} = p_{MH} \quad .$$

States 6, 7, and 8 are search states. State 8, "search (full field)," is equivalent to State 1 (entry) for the same reasons States 1 and 5 in the six-state model are equivalent, and the comments there apply here also. Specifically,

$$h^*_{8i}(t) = (n_D\lambda_D + n_H\lambda_H)e^{-(n_D\lambda_D + n_H\lambda_H)t}, \quad t \geq 0$$

and

$$p^*_{82} = \frac{n_D\lambda}{n_D\lambda_D + n_H\lambda_H} \quad \text{and} \quad p^*_{83} = \frac{n_H\lambda_H}{n_D\lambda_D + n_H\lambda_H} \quad .$$

As before, n_D is the number of decoys, λ_D the single decoy encounter rate,† λ_H the single HVS encounter rate,† and n_H the number of HVSs. States 6 and 7 are reduced field search states. State 6, "search (decoy reduced field)," is the same as State 4 in the six-state model. Hence,

---

†Discussed in Section 3.3.

$$h^*_{6i}(t) = [(n_D-1)\lambda_D + n_H\lambda_H + \lambda_R]e^{-[(n_D-1)\lambda_D+n_H\lambda_H+\lambda_R]t} \quad , \quad t \geq 0$$

for $n_D \geq 1$ and $i = 2, 3, 8$. Recall λ_R^{-1} is the mean information retention time. Similarly,

$$p^*_{62} = \frac{(n_D-1)\lambda_D}{(n_D-1)\lambda_D + n_H\lambda_H + \lambda_R}$$

$$p^*_{63} = \frac{n_H\lambda_H}{(n_D-1)\lambda_D + n_H\lambda_H + \lambda_R}$$

$$p^*_{68} = \frac{\lambda_R}{(n_D-1)\lambda_D + n_H\lambda_H + \lambda_R} \quad .$$

State 7, "search (NVS reduced field)," is analogous to State 6. Instead of n_D, n_H is reduced by one, so that

$$h^*_{7i}(t) = [n_D\lambda_D + (n_H-1)\lambda_H + \lambda_R]e^{-[n_D\lambda_D+(n_H-1)\lambda_H+\lambda_R]t} \quad , \quad t \geq 0$$

for $n_H \geq 1$ and $i = 2, 3, 8$ and

$$p^*_{72} = \frac{n_D\lambda_D}{n_D\lambda_D + (n_H-1)\lambda_H + \lambda_R}$$

$$p^*_{73} = \frac{(n_H-1)\lambda_H}{n_D\lambda_D + (n_H-1)\lambda_H + \lambda_R}$$

$$p^*_{78} = \frac{\lambda_R}{n_D\lambda_D + (n_H-1)\lambda_H + \lambda_R} \quad .$$

Finally, States 9 and 10 are dummy termination states. They are trapping states, so $p^*_{99} = 1$ and $p^*_{1010} = 1$, and the forms of $h^*_{99}(\cdot)$ and $h^*_{1010}(\cdot)$ are not relevant.

3. GEOMETRICAL ANALYSIS

This section presents the detailed derivation of certain quantities alluded to in the previous section on model structure. These derivations were delayed to this point so that the details would not obscure the basic model.

3.1 Transition Probabilities p_{2i} and p_{6i}

The first step here is to describe the transition probabilities in terms of events. To simplify the notation we will denote

$$P(\text{next event is A} \mid \text{last event was B})$$

simply by

$$P(A \mid B) \quad .$$

Then we have

p_{23} = P(classify an HVS | just classified 1st decoy)

p_{24} = P(resume search | just classified 1st decoy)

p_{26} = P(classify another decoy | just classified 1st decoy)

and

p_{63} = P(classify an HVS | just classified 2nd or later decoy)

p_{64} = P(resume search | just classified 2nd or later decoy)

p_{66} = P(classify another decoy | just classified 2nd or later decoy).

Here the phrase "just classified 1st decoy" refers to the first decoy encountered after a search period with no contacts and not necessarily to the first decoy encountered during the game. Several "1st decoys" may occur during any given time period. Similarly, "2nd or later decoy" refers to any decoy in a string initiated by a 1st decoy, where a "string" means the submarine goes immediately from one contact to another without an interval of "no contact."

In computing these transition probabilities we use somewhat the same approach as used in Ref. 3 to compute the α_{2i}'s. However, one of the crucial independence assumptions used there is shown here to be unnecessary.

Consider an area of A_o square nautical miles containing n_D decoys, n_H HVSs, and one searcher. Assume that at any given instant the decoys and HVSs are uniformly distributed at random over the area. The completion of classification of a decoy anticipates an event. The next event to occur can be either start of search, start of classification of (another) decoy, or start of classification of an HVS. In order for the next event to be start of classification of an HVS, at least one HVS must be present* and, if one or more decoys are also present, an HVS must be chosen from the array of targets. In particular

$$p_{23} = P(\text{classify an HVS} \mid \text{just classified 1st decoy})$$

$$= P(\text{at least one HVS present and pick HVS from field of available targets} \mid \text{just classified 1st decoy})$$

*"Present" means within detection range of submarine.

$$p_{23} = \sum_{k=0}^{n_D-1} \sum_{\ell=1}^{n_H} P(k \text{ new decoys present, } \ell \text{ HVSs present, pick HVS} \mid \text{just classified 1st decoy})$$

$$= \sum_{k=0}^{n_D-1} \sum_{\ell=1}^{n_H} P(\text{pick HVS} \mid k \text{ new decoys, } \ell \text{ HVSs, just classified 1st decoy})$$

$$P(\ell \text{ HVS} \mid k \text{ new decoys, just finished 1st decoy})$$

$$P(k \text{ new decoys} \mid \text{just finished 1st decoy})$$

As in Ref. 3 we assume

$$P(\text{pick HVS} \mid k \text{ new decoys, } \ell \text{ HVSs, just classified 1st decoy})$$

$$= Pr(\text{pick HVS} \mid k \text{ new decoys, } \ell \text{ HVSs})$$

$$= \frac{\ell}{\ell + k}$$

and also that

$$P(k \text{ new decoys} \mid \text{just classified 1st decoy})$$

$$= P(k \text{ new decoys within radius } R_D \text{ of submarine} \mid \text{just classified 1st decoy})$$

$$= \binom{n_D-1}{k} P_{D2}^{k} (1 - P_{D2})^{n_D-1-k}$$

where P_{D2} = P (any one decoy is within radius R_D of submarine|just classified 1st decoy). The binomial form assumes stochastic independence of the decoys. In previous work $P_D = \pi R_D^2/A_o$ was used. A better approach is developed as follows.

P_{D2} is given in definitional form by

$$P_{D2} = \sum_{i=1}^{n_r} \sum_{j=1}^{n_\varphi(r_i)} \frac{w(r_i)}{n_\varphi(r_i)} \frac{B_D(r_i) - B_1(r_i,\varphi_j)}{A_o}$$

where

i = range index

j = angle index

r_i = a range from center of operating area

φ_j = an orientation angle

n_r = number of range increments

$n_\varphi(r_i)$ = number of angle increments at range r_i.

This formulation is derived in the paragraphs below. Let

E = the event: any one decoy is within radius R_D of submarine

C_1 = the event: submarine just classified 1st decoy

r_i = the event: submarine is located distance r_i from center of operating area

φ_j = the event: submarine position at start of classification is located at angle φ_j from submarine position at end of classification (see Figure 3.1; measured counterclockwise from radius).

Then

$$P(E|C_1) = \sum_i {}' \sum_j P(E, r_i, \varphi_j | C_1)$$

from the law of total probability. Further,

$$P(E, r_i, \varphi_j | C_1) = P(E | r_i, \varphi_j, C_1) \cdot P(\varphi_j | r_i, C_1) \cdot P(r_i | C_1) \quad .$$

To obtain $P(r_i|C_1)$ we assume the submarine position is uniformly distributed throughout A_o, given that C_1 occurs. It is shown in Appendix C that the distance r from the center of A_o has density function

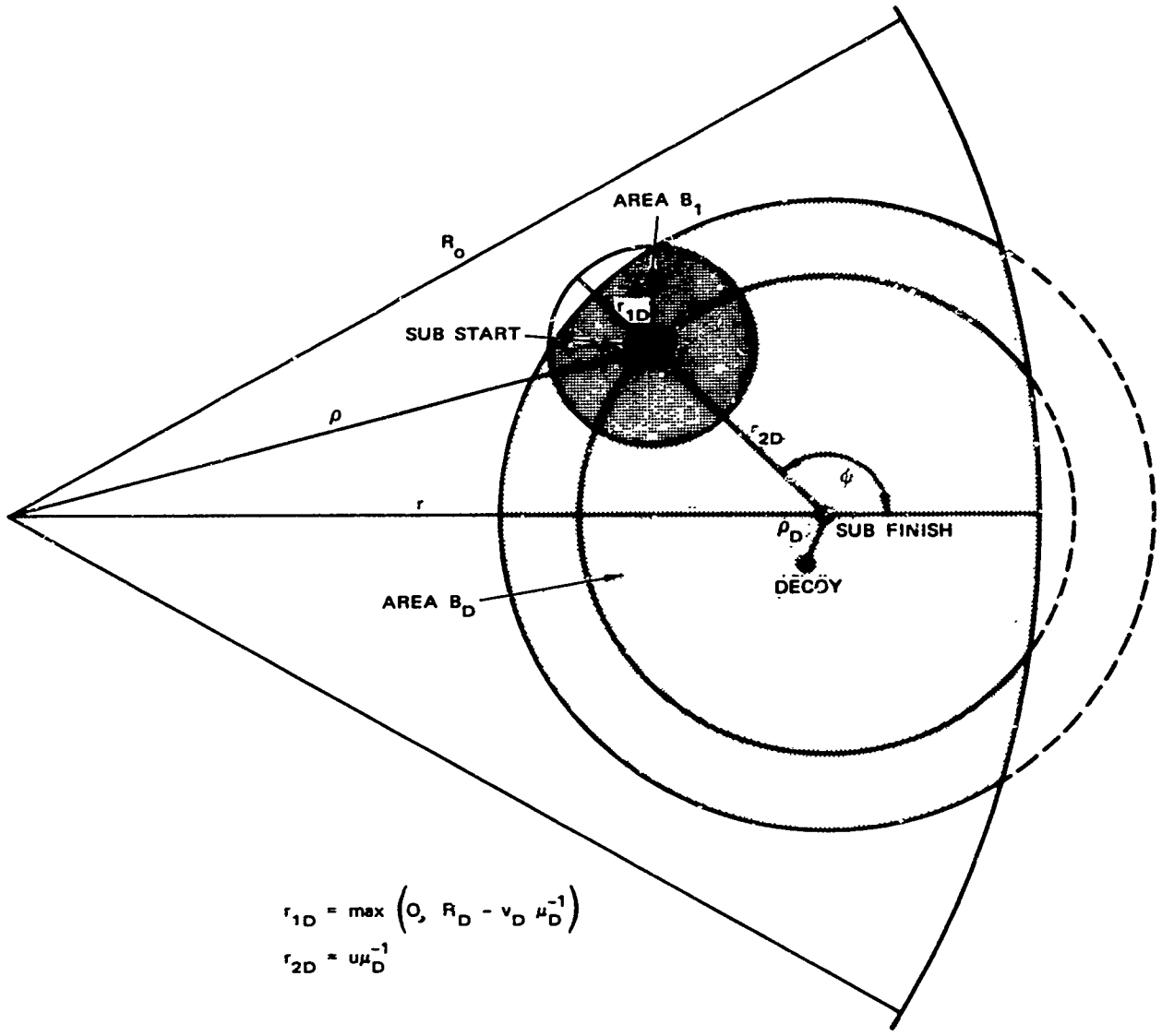


FIGURE 3.1 GEOMETRY AT END OF DECOY CLASSIFICATION

$$k(r) = \frac{2\pi}{A_o} r \qquad 0 \le r \le R_o \quad .$$

We consider a discrete approximation in 20-nmi increments starting with distance 10 nmi, so that

$$w(r_1) = 100\pi/A_o$$

$$w(r_i) = [400 + 800(i-1)]\pi/A_o \qquad i = 2, \ldots, N_R$$

where N_R = integer part of

$$R_o/20 + 0.5 \quad \text{and} \quad \{r_1, r_2, r_3, \ldots\} = \{10, 30, 50, \ldots\} \quad .$$

The density and discrete approximation are illustrated below in Figure 3.2 for an operating area of radius 200 nmi. Then, $P(r_i|C_i) = w(r_i)$.

The angle φ is assumed uniformly distributed within the operating area and a discrete approximation taken. Symmetry is taken into account to reduce calculations. The number of elements in the discrete

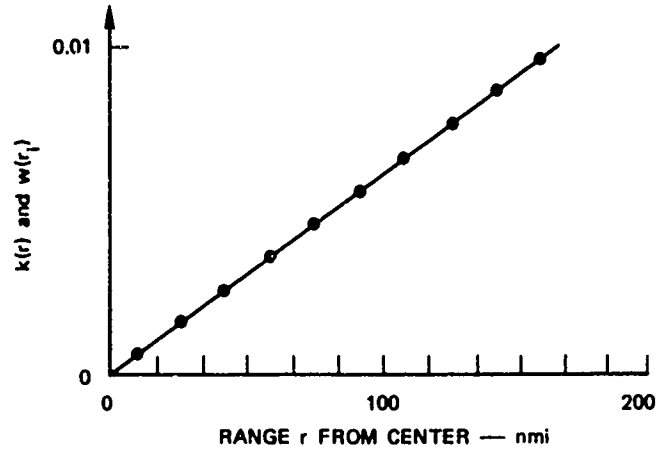


FIGURE 3.2 RANGE DENSITY AND APPROXIMATION

approximation depends on the range r_i from the center and is denoted $n_\varphi(r_i)$. The angle increment is 30°, with the first angle at 15°, giving possible angles {15°, 45°, 75°, 105°, 135°, 165°}. Symmetry accounts for angles between 180° and 360°. Only angles such that the distance $r_{2D} = u\mu_D^{-1}$ is entirely inside* A_o are included in $n_\varphi(r_i)$. Recall μ_D^{-1} is mean decoy capture (or classification) time and u is submarine speed. Hence, the distance r_{2D} is a measure of travel by the submarine while captured by the decoy. So

$$P(\varphi_j | r_i, C_1) = \frac{1}{n_\varphi(r_i)} \quad .$$

It remains to calculate $P(E|r_i, \varphi_j, C_1)$. This probability is taken to be

$$P(E|r_i,\varphi_j,C_1) = \frac{B_D(r_i) - B_1(r_i,\varphi_j)}{A_o} \quad ,$$

where

A_o = size of operating area in sq nmi (this symbol alternately refers to the geometrical figure as well as the number of sq nmi)

$B_D(r_i)$ = the intersection of A_o and a circle of radius R_D centered on a radial at distance r_i from the center of A_o (the submarine is at r_i)

$B_1(r_i, \varphi_j)$ = the intersection of $B_D(r_i)$ and a circle of radius r_{1D} centered at a distance r_{2D} and angle φ_j from the center of $B_D(r_i)$,

with

$$r_{1D} = \max [0, R_D - v_D\mu_D^{-1}] \quad .$$

---

*Inside A_o if $r_i^2 + r_{2D}^2 + 2r_i r_{2D} \cos \varphi_j \leq R_o^2$.

Here r_{1D} is the radius of the (perhaps only) partially dissipated swept area present at the initiation of 1st decoy classification. The quantity μ_D is decoy speed. The computation of $B_D(r_i)$ and $B_1(r_1, \varphi_j)$ involves r_{1D}, r_{2D}, R_o and R_D, as well as r_i and φ_j; these areas, distances, and angles are shown in Figure 3.1.

It is useful here to introduce the function $A_F(r, R_1, R_2)$, which gives the overlap area of two circles of radii R_1 and R_2 whose centers are a distance r apart. See Figure 3.3. First define

$$\theta_1 = \pi - \arccos\left[\frac{R_2^2 + r^2 - R_1^2}{2rR_2}\right]$$

$$\theta_2 = \arccos\left[\frac{R_1^2 + r^2 - R_2^2}{2rR_1}\right] .$$

Then

$$A_F(r,R_1,R_2) = \begin{cases} 0 & \text{if} \quad r > R_1 + R_2 \\ \pi\,[\min(R_1,R_2)]^2 & \text{if} \quad r \le |R_1 - R_2| \\ \pi R_2^2 - \frac{R_2^2}{2}[2\theta_1 - \sin 2\theta_1] & \\ \quad + \frac{R_1^2}{2}[2\theta_2 - \sin 2\theta_2] & \text{if} \quad r > |R_1 - R_2| \end{cases} .$$

Thus, $B_D(r_i) = A_F(r_i, R_o, R_D)$. However, $B_1(r_i, \varphi_j)$ is more involved, including some numerical integration. The computation is given in flow chart form in Figure 3.4. This flow chart is in generic form, i.e., it

describes a quantity useful for either decoy or HVS. When entered with R_D, r_{1D} and r_{2D}, the output is $B_1(r_i, \varphi_j)$.

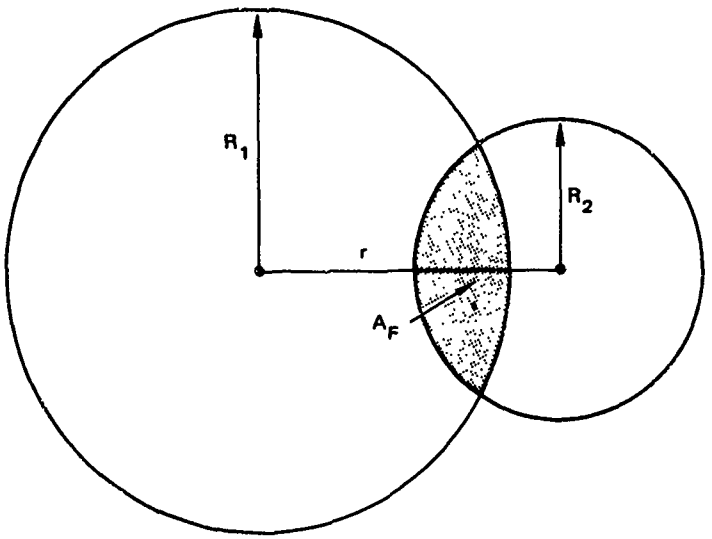


FIGURE 3.3 OVERLAP AREA A_F

A comparison between this more involved form of P_{D2} with the previous approximation of $\pi R_D^2/A_o$ is shown in Figure 3.5. In this figure P_{D2} is shown as a function of R_D and μ_D^{-1}, assuming A_o is 200 nmi in radius. It is seen here that the previous approximation for an R_D of 200 nmi differs by a factor of about 20 from the new form with mean capture time of 1 hour and by a factor of 2 with a mean capture time of 10 hours. On the other hand, for larger capture times and $R_D \leq 50$ nmi, the previous approximation is good. Numerical experiments have shown that $P_{D2} \to 0$ for large values of R_D (greater than the radius of A_o). This happens because of the inherent assumption of a previous condition of no contact. Algebraically what happens as $R_D \to \infty$ is $r_{2D} \ll r_{1D} \approx R_D$ which implies $B_D - B_1 \approx 0$. Operationally this implies a predominance of swept area hanging over from the initial assumption of start without contacts which is unreasonable for very large R_D (relative to A_o).

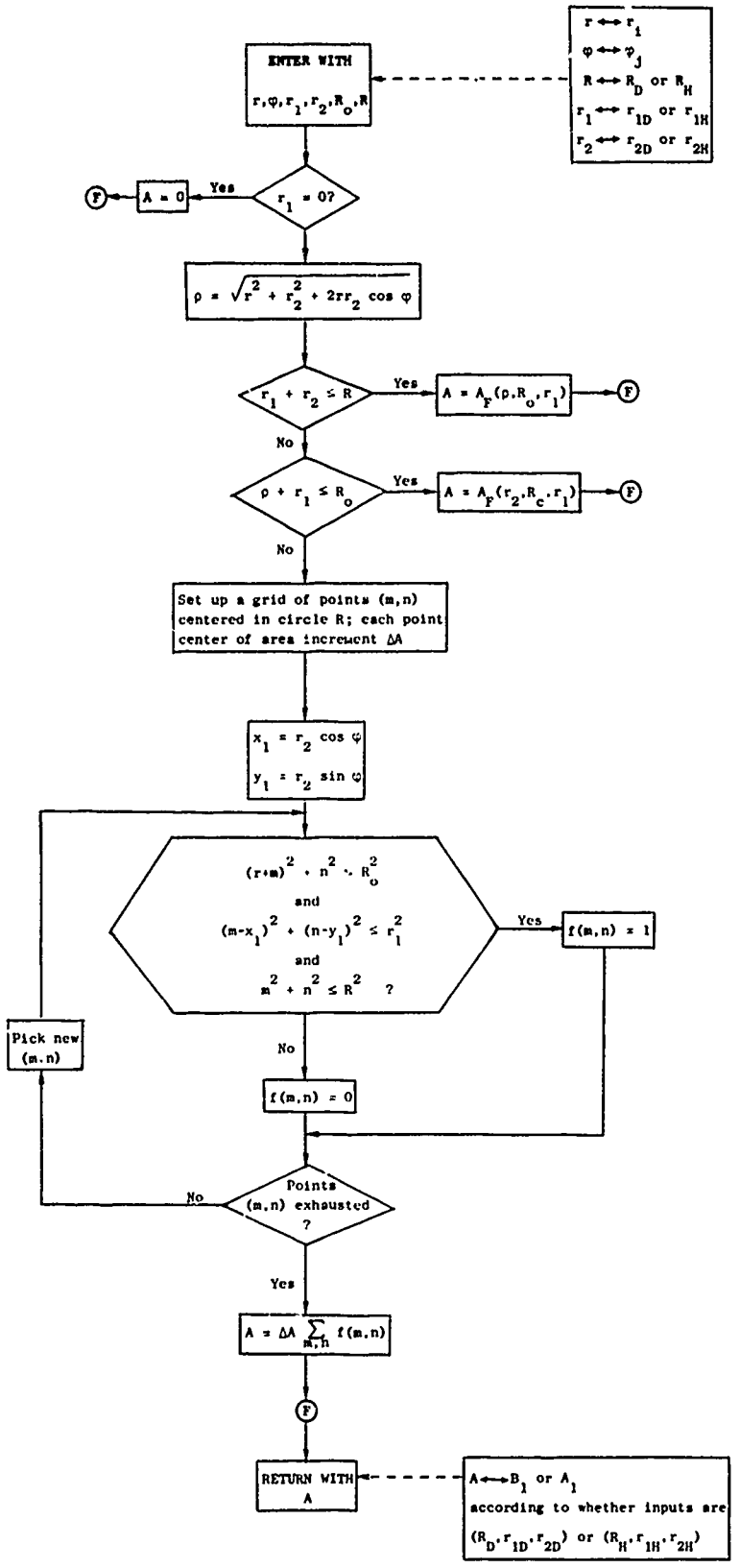


FIGURE 3.4 CALCULATION OF REMAINING SWEPT AREA

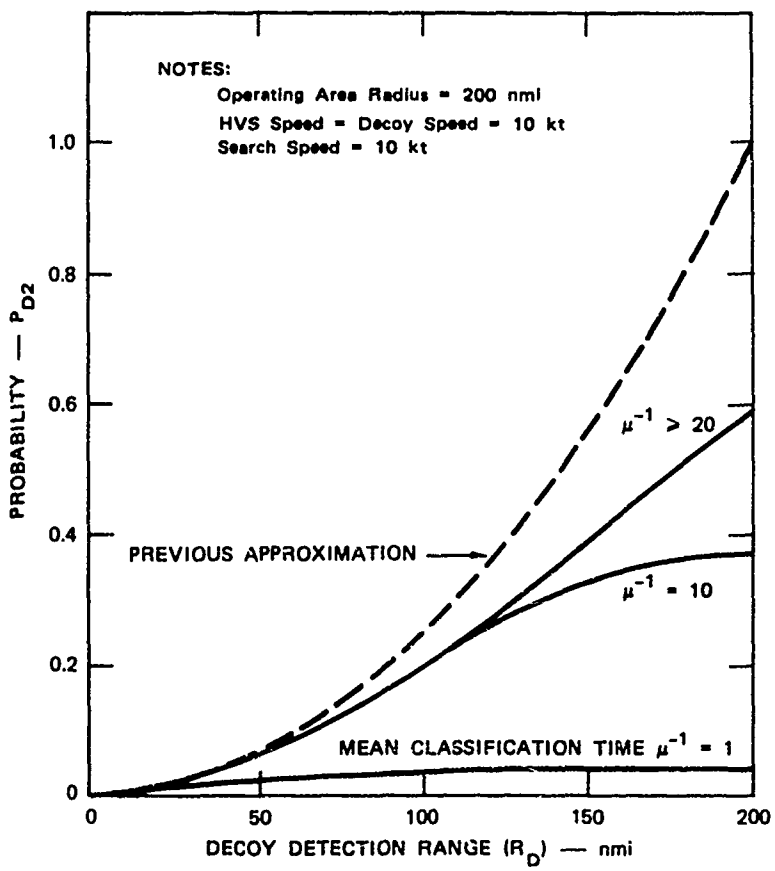


FIGURE 3.5 PROBABILITY P_{D2} VERSUS DECOY DETECTION RANGE (R_D)

Having established P_{D2} we are finished with the probability

$$P(\text{k new decoys} \mid \text{just classified 1st decoy}) \quad .$$

In order to complete the derivation of p_{23}, the remaining conditional probability

$$Pr(\ell \text{ HVSs} \mid \text{k new decoys, just finished 1st decoy})$$

is now developed. We will show this probability is actually independent of k. First consider k = 1, i.e.,

$$P(\text{1 HVS} \mid \text{1 new decoy, just finished 1st decoy}) \quad .$$

It is convenient to drop the "1" from "1 HVS" and just consider the situation of a single HVS in A_o; this is consistent with our later treatment of several independent HVSs. Assume

$$P(\text{HVS}\,|\,\text{1 new decoy, just classified 1st decoy})$$

$$= P(\text{HVS}\,|\,\text{2 decoys}) \quad .$$

This assumption can be interpreted as saying HVS movement or position is independent of the submarine's state with respect to the decoys. We can write P(HVS|2 decoys) the same way we wrote P_{D2}, i.e.,

$$P(\text{HVS}\,|\,\text{2 decoys}) = P(H|F_2)$$

$$= \sum_i \sum_j P(H|r_i,\varphi_j,F_2) \cdot P(\varphi_j|r_i,F_2) \cdot P(r_i|F_2)$$

where

H = the event: HVS is within radius R_H of submarine

F_2 = the event: 2 decoys within R_D of submarine, one just classified, one new

anu r_i and φ_j are defined as before. F_2 plays the role here that C_1 played for P_{D2}, so we have

$$P(\varphi_j|r_i,F_2) = \frac{1}{n_\varphi(r_i)}$$

and

$$P(r_i|F_2) = w(r_i) \quad .$$

$P(H|r_i, \varphi_j, F_2)$ is treated similarly to $P(E|r_i, \varphi_j, C_1)$, except we wish to show "H" is independent of "F_2," i.e., independent of the number of decoys present.

Examine Figure 3.6. In this figure $A_H(r_i)$ is: the intersection of A_o and a circle of radius R_H (HVS detection range) centered at the

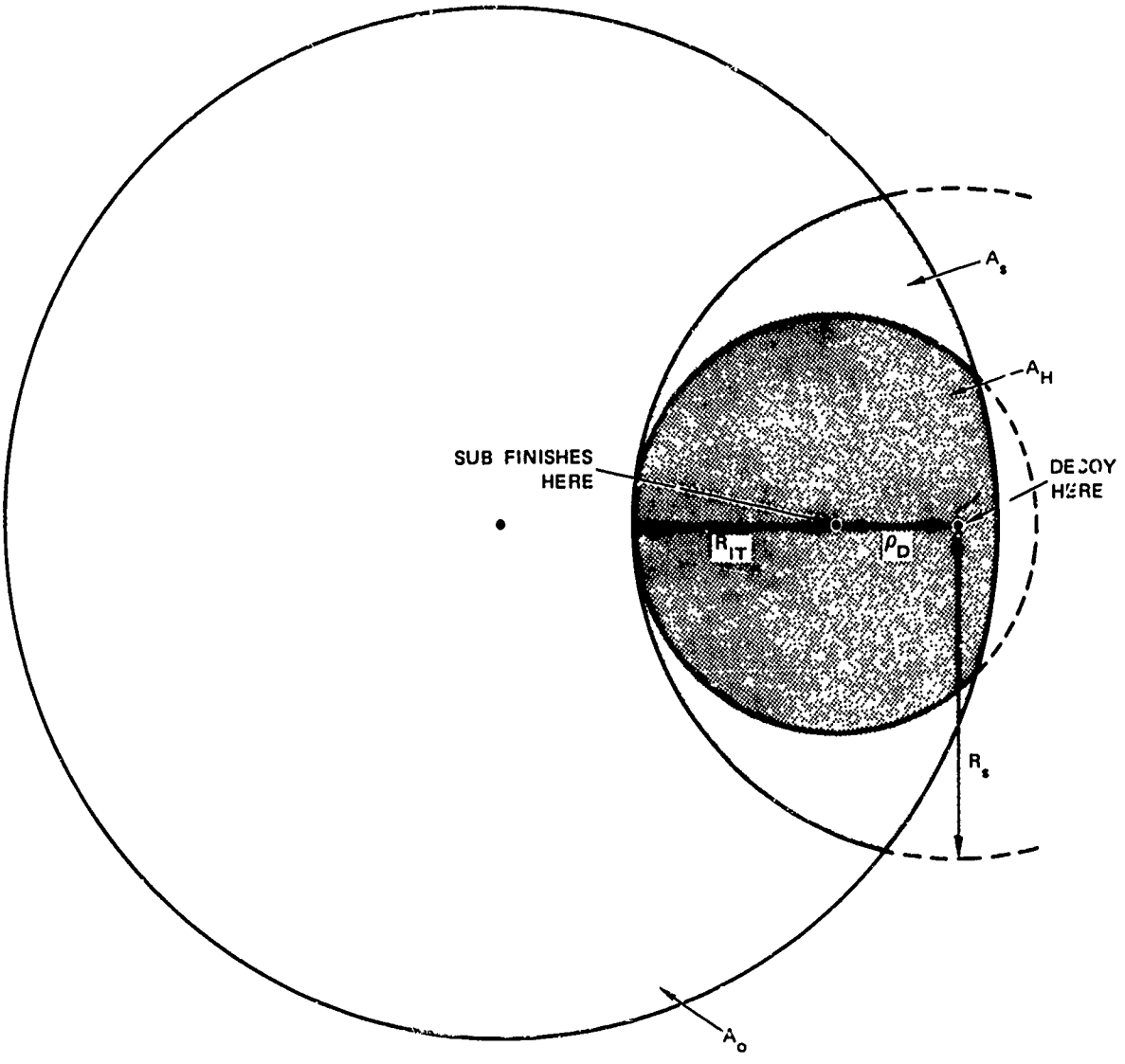


FIGURE 3.6 ILLUSTRATION OF AREAS A_H AND A_s

submarine (located r_i from center of A_o) at the time of completion of a 1st decoy classification. A_H is the HVS analogue of the decoy quantity B_D, i.e., $A_H(r_i) = A_F(r_i, R_o, R_H)$. A_s is the intersection of A_o and a circle of radius $R_s = \rho_D + R_H$ centered at the decoy shown in the figure,

where ρ_D is decoy classification rarge.* Each decoy has an A_s area associated with it. However, to keep notation simple we avoid as far as possible indexing A_s with respect to decoys. Note $A_H \subset A_s$ each A_s. Not shown explicitly in Figure 3.6 but part of the analysis is area A_1, which is the intersection of $A_H(r_i)$ and a circle of radius r_{1H} centered at a distance r_{2H} and angle φ_j from the center of $A_H(r_i)$, where

$$r_{1H} = \max [0, R_H - v_H \mu_H^{-1}]$$

and

$$r_{2H} = u\mu_H^{-1} \quad .$$

A_1 as the HVS analogue of the decoy B_1 and its calculation is shown in Figure 3.6.

Now we are interested in computing $P(H| r_i, \varphi_j, F_2)$. For convenience we temporarily drop r_i and φ_j and rewrite "F_2" as "2 decoys ϵB_D," so

$$P(H|F_2) = P(H|2 \text{ decoys } \epsilon B_D) \quad .$$

In order to derive this quantity we have considered some geometrical concepts. We need also the concepts related to overlap of influence between HVS and decoys. In Ref. 3 a queuing model of these phenomena was introduced. Let

λ_c = rate at which HVS encounters decoys (discussed in Section 3.3)

λ_o = maximum acceptable rate of HVS course change for purpose of decoy avoidance.

---

*This range plays a minor role in the model; much more important in the model context is decoy classification time.

For example, an HVS may be subject to constraints such that it would be willing to change course to avoid a decoy on the average of only once in every 5 hours, in which case $\lambda_o = 0.2\ hr^{-1}$. We can now model the situation of a single HVS as a single server queue with customer balking, that is, when the customer arrives if the server is busy the arriving customer leaves. In this case the HVS is the server and the customer is a decoy. The customer arrival rate is λ_c, and the service rate is λ_o. Customer arrival corresponds to HVS contact of a decoy. Customer service corresponds to the length of time after a decoy avoidance maneuver before the HVS is willing to make another such maneuver. Such a queuing system can be represented by a two-state Markov process, where the States 0 or 1 correspond to the number of customers being served or, equivalently, whether or not the server is busy. Let B denote "system in State 1" and $\bar{B}$ "system in State 0." Assume the system is in steady state. Then, from Ref. 3 we have

$$P(B) = \frac{\lambda_c}{\lambda_o + \lambda_c}$$

$$P(\bar{B}) = \frac{\lambda_o}{\lambda_o + \lambda_c} \quad .$$

P(B) is interpreted as the probability the HVS is unwilling to maneuver to avoid a decoy at any given time.

With States B and $\bar{B}$ defined we can write

$$P(H \mid 2 \text{ decoys } \epsilon B_D) =$$

$$P(H \mid B \ \& \ 2 \text{ decoys } \epsilon B_D)\ P(B \mid 2 \text{ decoys } \epsilon B_D)$$

$$+ P(H \mid \bar{B} \ \& \ 2 \text{ decoys } \epsilon B_D)\ P(\bar{B} \mid 2 \text{ decoys } \epsilon B_D) \quad .$$

We have

$$P(H|\bar{B} \text{ \& 2 decoys } \epsilon B_D) = 0$$

because "H" implies "B" since $A_H \subset A_s$. That is, an HVS would be in A_H only if it had chosen to overlap with a decoy and hence was in State "B" of the overlap model.

Examine Figure 3.7. The area A^* is the union of $A_s^{(1)}$ and $A_s^{(2)}$, where the latter areas correspond to decoys D_1 and D_2, respectively. Now write

$$\begin{aligned} &P(H|B \text{ \& 2 decoys } \epsilon B_D) \\ &= P(H|HVS\epsilon A^* \text{ \& } B \text{ \& 2 decoys } \epsilon B_D) \\ &\quad \cdot Pr(HVS\epsilon A^*|B \text{ \& 2 decoys } \epsilon B_D) \\ &\quad + P(H|HVS\notin A^* \text{ \& } B \text{ \& 2 decoys } \epsilon B_D) \\ &\quad \cdot P(HVS\notin A^*|B \text{ \& 2 decoys } \epsilon B_D) \quad . \end{aligned}$$

As above, $P(H|HVS\notin A^* \text{ \& } B \text{ 2 decoys } \epsilon B_D) = 0$ since $A_H \subset A^*$. Further, assume

$$P(H|HVS\epsilon A^* \text{ \& } B \text{ \& 2 decoys } \epsilon B_D) = \frac{A_H - A_1}{A^*}$$

and

$$P(HVS\epsilon A^*|B \text{ \& 2 decoys } \epsilon B_D) = \frac{A^*}{A_o}$$

so

$$P(H|B \text{ \& 2 deco} \cdot \epsilon B_D) = \frac{A_H - A_1}{A_o} \quad .$$

Here we have assumed independence of "$HVS\epsilon A^*$" and "H" from "2 decoys ϵB_D" when the former are also conditioned on "B." In addition, assume $P(B|2 \text{ decoys } \epsilon B_D) = P(B)$, so that we finally have

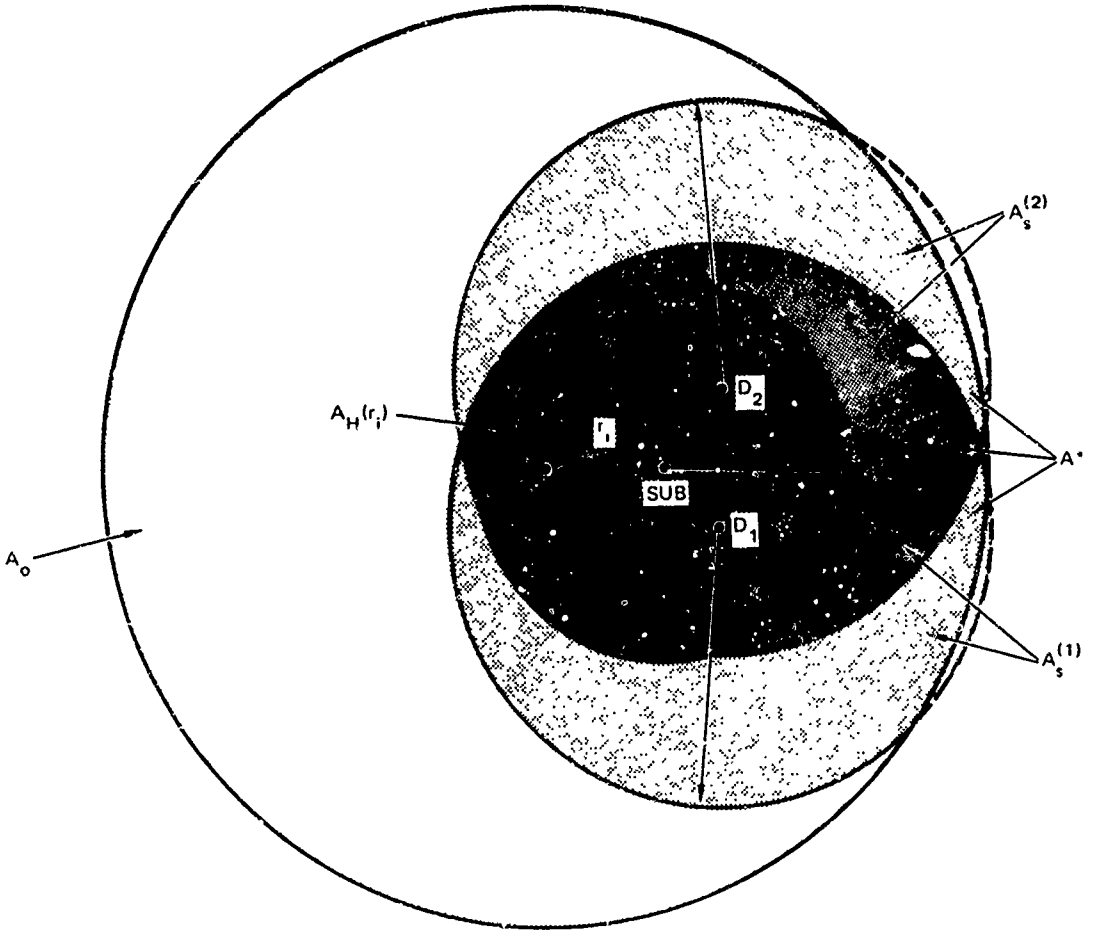


FIGURE 3.7 AREAS IN QUEUE MODEL OF INFLUENCE OVERLAP

$$P(H \mid 2 \text{ decoys } \epsilon B_D) = \frac{A_H - A_1}{A_o} P(B) \quad .$$

But this indicates "H" is independent of "2 decoys ϵB_D" regardless of conditioning on "B." Furthermore, it is evident that the same is true for "n decoys ϵB_D" all n = 1, 2, ..., so in general

$$P(H) = \frac{A_H - A_1}{A_o} \frac{\lambda_c}{\lambda_o + \lambda_c}$$

or in terms of our earlier notation

$$P(H|r_i\varphi_j,F_2) = P(H|r_i,\varphi_j) = \frac{A_H(r_i) - A_1(r_i,\varphi_j)}{A_o} \cdot \frac{\lambda_c}{\lambda_o + \lambda_c} \quad .$$

Retreating back through all the notation we can now express

$$P(\text{HVS}|\text{k new decoys, just classified 1st decoy})$$

$$= \frac{\lambda_c}{\lambda_o + \lambda_c} \sum_i \sum_j \frac{w(r_i)}{n_\varphi(r_i)} \frac{A_H(r_i) - A_1(r_i,\varphi_j)}{A_o}$$

$$\triangleq P_{H2}$$

In particular, "HVS" is independent of k, the number of new decoys. Assuming the HVSs are stochastically independent, we have

$$P(\ell\text{HVSs}|\text{k new decoys \& just classified 1st decoy})$$

$$= \binom{n_H}{\ell} P_{H2}^{\ell}(1 - P_{H2})^{n_H-\ell}$$

thus

$$P_{23} = \sum_{k=0}^{n_D-1} \sum_{\ell=1}^{n_H} \frac{\ell}{\ell + k} \binom{n_D-1}{k} P_{D2}^{k}(1 - P_{D2})^{n_D-1-k} \binom{n_H}{\ell} P_{H2}^{\ell}(1 - P_{H2})^{n_H-\ell} \quad .$$

To obtain P_{24} we simply note

$$P_{24} = P(\text{resume search}|\text{just classified 1st decoy})$$

$$= P(\text{0 HVS and 0 new decoys present}|\text{just classified 1st decoy})$$

$$= P(\text{0 HVS present}|\text{just classified 1st decoy})$$

$$\cdot P(\text{0 new decoys present}|\text{just classified 1st decoy})$$

$$= [1 - P_{H2}]^{n_H}[1 - P_{D2}]^{n_D-1} \quad .$$

Finally, by the law of total probability

$$p_{26} = 1 - p_{23} - p_{24} \quad .$$

State 6 is analogous to State 2, with the exception that the areas A_1 and B_1 are assumed to be zero, i.e., the swept area is dissipated by the time the 2nd or later decoy in the sequence is encountered. Specifically,

$$P_{D6} = \sum_{i=1}^{n_r} \sum_{j=1}^{n_\varphi(r_i)} \frac{w(r_i)}{n_\varphi(r_i)} \frac{B_D(r_i)}{A_o}$$

$$P_{H6} = \sum_{i=1}^{n_r} \sum_{j=1}^{n_\varphi(r_i)} \frac{w(r_i)}{n_\varphi(r_i)} \frac{A_H(r_i)}{A_o}$$

so

$$p_{63} = \sum_{k=0}^{n_D-1} \sum_{\ell=1}^{n_H} \frac{\ell}{\ell + k} \binom{n_D-1}{k} P_{D6}^{k} (1 - P_{D6})^{n_D-1-k} \binom{n_H}{\ell} P_{H6}^{\ell} (1 - P_{H6})^{n_H-\ell}$$

$$p_{64} = [1 - P_{H6}]^{n_H} [1 - P_{D6}]^{n_H}$$

$$p_{66} = 1 - p_{63} - p_{64} \quad .$$

3.2 Probabilities p'_{3i} and p'_{5i}

The probabilities p'_{3i} and p'_{5i} are used in computing transition probabilities p^*_{3i} and p^*_{5i}. In a sense p'_{3i} and p'_{5i} are pseudotransition

probabilities. Strictly speaking, the p'_{3i} and p'_{5i} are complete sets of transition probabilities in that $\sum_i p'_{3i} = 1$ and $\sum_i p'_{5i} = 1$; they are augmented, however, with p_{MH} to obtain p^*_{3i} and p^*_{5i}, as described in Section 2. The geometrical situations for p'_{3i} and p'_{5i} are identical to those for p_{2i} and p_{6i}, respectively. The difference in the situations is that, for the former, an HVS has just been classified, while in the latter a decoy has just been classified. This difference merely requires the substitutions of n_D for $n_D - 1$ and $n_H - 1$ for n_H in the computations for p_{2i} and p_{6i} in order to obtain p'_{3i} and p'_{5i}. Hence, we have

$$p'_{34} = \sum_{k=1}^{n_D} \sum_{\ell=0}^{n_H-1} \frac{k}{\ell + k} \binom{n_D}{k} P_{D2}^{k} (1 - P_{D2})^{n_D - k} \binom{n_H - 1}{\ell} P_{H2}^{\ell} (1 - P_{H2})^{n_H - 1 - \ell}$$

$$p'_{37} = [1 - P_{H2}]^{n_H - 1} [1 - P_{D2}]^{n_D}$$

$$p'_{35} = 1 - p'_{34} - p'_{37}$$

and

$$p'_{54} = \sum_{k=1}^{n_D} \sum_{\ell=0}^{n_H-1} \frac{k}{\ell + k} \binom{n_D}{k} P_{D6}^{k} (1 - P_{D6})^{n_D - k} \binom{n_H - 1}{\ell} P_{H6}^{\ell} (1 - P_{H6})^{n_H - 1 - \ell}$$

$$p'_{57} = [1 - P_{H6}]^{n_H - 1} [1 - P_{D6}]^{n_D}$$

$$p'_{55} = 1 - p'_{54} - p'_{57}$$

where P_{D2}, P_{D6}m P_{H2}, and P_{H6} were defined in the preceding subsection.

3.3 Detection Rates

In developing the model structure, reference was made to random variables

T_D = time to detect a decoy

T_H = time to detect an HVS

with pdf's

$$f_H = \beta_H e^{-\beta_H t} \quad , \quad t \geq 0 \quad ,$$

$$f_D = \beta_D e^{-\beta_D t} \quad , \quad t \geq 0 \quad .$$

Also in discussing transition probabilities, reference was made to rate λ_c, the rate at which an HVS encounters decoys. These are three entities with the common feature of respresenting Poisson processes, that is, the time between detections (or encounters) has the exponential distribution. The rates corresponding to these processes are β_H, β_D, and λ_c. It is characteristic of the Poisson process with rate λ that, if n is the number of occurrences in time Δt, then

$$p(n=0;\Delta t) = 1 - \lambda \cdot \Delta t + o(\Delta t)$$

$$p(n=1;\Delta t) = \lambda \cdot \Delta t + o(\Delta t) \quad .$$

We assume the rate λ is dependent on r_i and φ_j, where r_i and φ_j are the range and heading angle discussed in Subsection 3.2. In this case

$$p(n=0;\Delta t) = \sum_i \sum_j p(n=0|r_i,\varphi_j;\Delta t)p(r_i)p(\varphi_j)$$

$$= \sum_i \sum_j [1 - \lambda(i,j)\Delta t + o(\Delta t)]p(r_i)p(\varphi_j)$$

$$= 1 - \left[\sum_i \sum_j \lambda(i,j)p(r_i)p(\varphi_j)\right]\Delta t + o(\Delta t) \quad .$$

That is,

$$\lambda = \sum_i \sum_j \lambda(i,j)p(r_i)p(\varphi_j) \quad ,$$

where $\lambda(i, j)$ is the rate obtained at range r_i and heading angle φ_j. The conditional rate $\lambda(i, j)$ is obtained from results contained in Ref. 6, explained as follows.

Consider a general target with speed v and detection range R. The submarine's speed is u on a heading φ_j and it is located a distance r_i from the center of A_o. It is shown in Ref. 6 that the rate at which targets are encountered at bearing β off the submarine heading is given by

$$\lambda(\beta;i,j) = \frac{\delta_i R}{\pi} f(\beta;u,v) \quad ,$$

where δ_i is the target density with submarine at r_i and for $v \leq u$

$$f(\beta;u,v) = v \cos^{-1}\left(-\frac{v}{u}\cos\beta\right)\cos\beta + \sqrt{u^2 - v^2\cos^2\beta}$$

and for $v > u$

$$f(\beta;u,v) = \begin{cases} \pi\, v \cos\beta & \text{when } -\cos^{-1}\frac{u}{v} \le \beta \le \cos^{-1}\frac{u}{v} \\ v\cos^{-1}\left(-\frac{v}{u}\cos\beta\right) + \sqrt{u^2 - v^2\cos^2\beta} & \text{when } -\cos^{-1}\left(-\frac{u}{v}\right) \le \beta \le -\cos^{-1}\left(\frac{u}{v}\right) \\ & \text{or when } \cos^{-1}\left(\frac{u}{v}\right) \le \beta \le \cos^{-1}\left(-\frac{u}{v}\right) \\ 0 & \text{when } \beta \le -\cos^{-1}\left(-\frac{u}{v}\right) \text{ or } \beta \ge \cos^{-1}\left(-\frac{u}{v}\right) \end{cases} .$$

These expressions assume an unbounded area. The effect of a bounded area such as we are dealing with is to set $\lambda(\beta;i, j) = 0$ when the point on the detection circle at heading β is outside A_o, since target density δ is assumed to be zero outside A_o. The total rate $\lambda(i, j)$ for a submarine on heading φ_j at range r_i is obtained by integrating $\lambda(\beta;i, j)$ with respect to β over the applicable range of β. To help visualize the range of β an example case is shown in Figure 3.8. In this example $\lambda(i, j)$ is given by

$$\lambda(i,j) = \int_o^{\beta_1} \lambda(\beta;i,j)d\beta + \int_o^{\beta_2} \lambda(\beta;i,j)d\beta \quad .$$

There are a number of separate geometrical cases that require individual attention in determining $\lambda(i, j)$. The logic and computations to handle these cases are shown in flow chart form in Figure 3.9. Note that in this flow chart, to conserve space, we denote

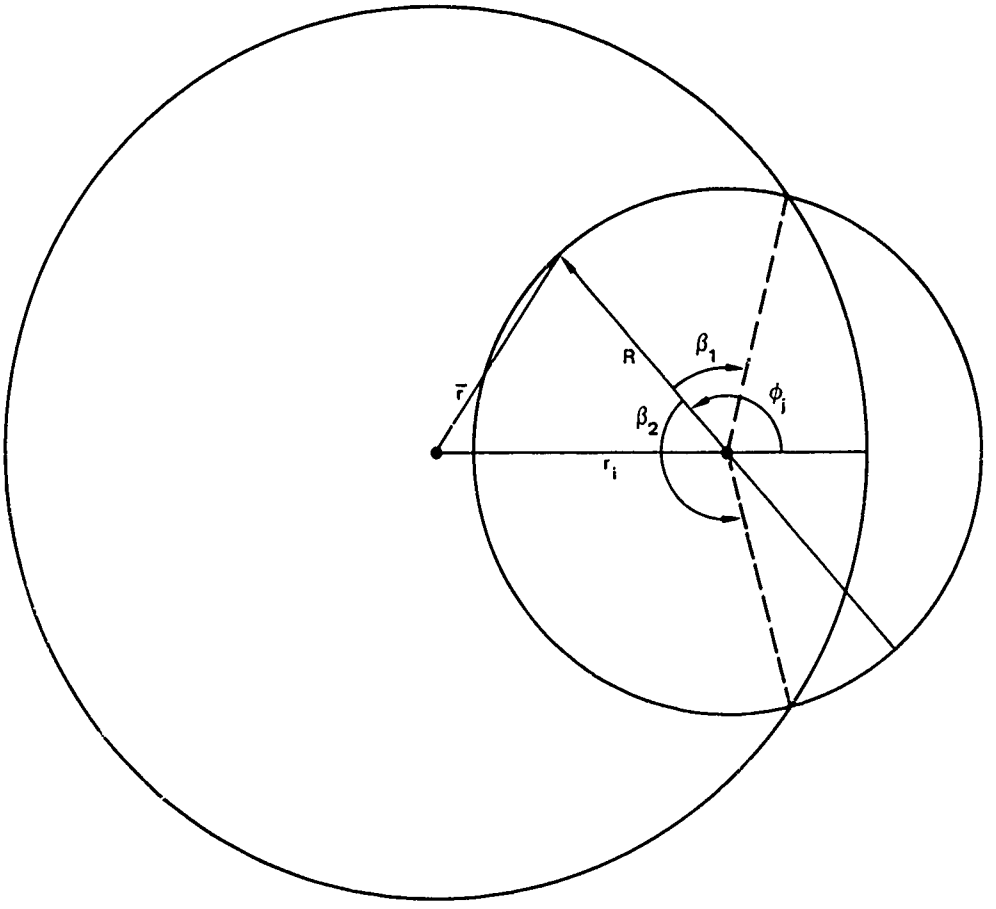


FIGURE 3.8 SAMPLE GEOMETRY FOR DETERMINING DETECTION RATE

$$\int_a^b \lambda(\beta; i, j)d\beta \text{ by } \int_a^b \quad .$$

Furthermore, let n_T denote the number of targets. Then

$$\delta_i = \frac{n_T}{A_o - A_T(r_i)} \quad ,$$

where $A_T(r_i)$ is, depending on target type, either $A_H(r_i)$ or $B_D(r_i)$ as discussed earlier. If we denote δ_i' to be density when $n_T = 1$, the $\delta_i = n_T\delta_i'$ and it is clear that

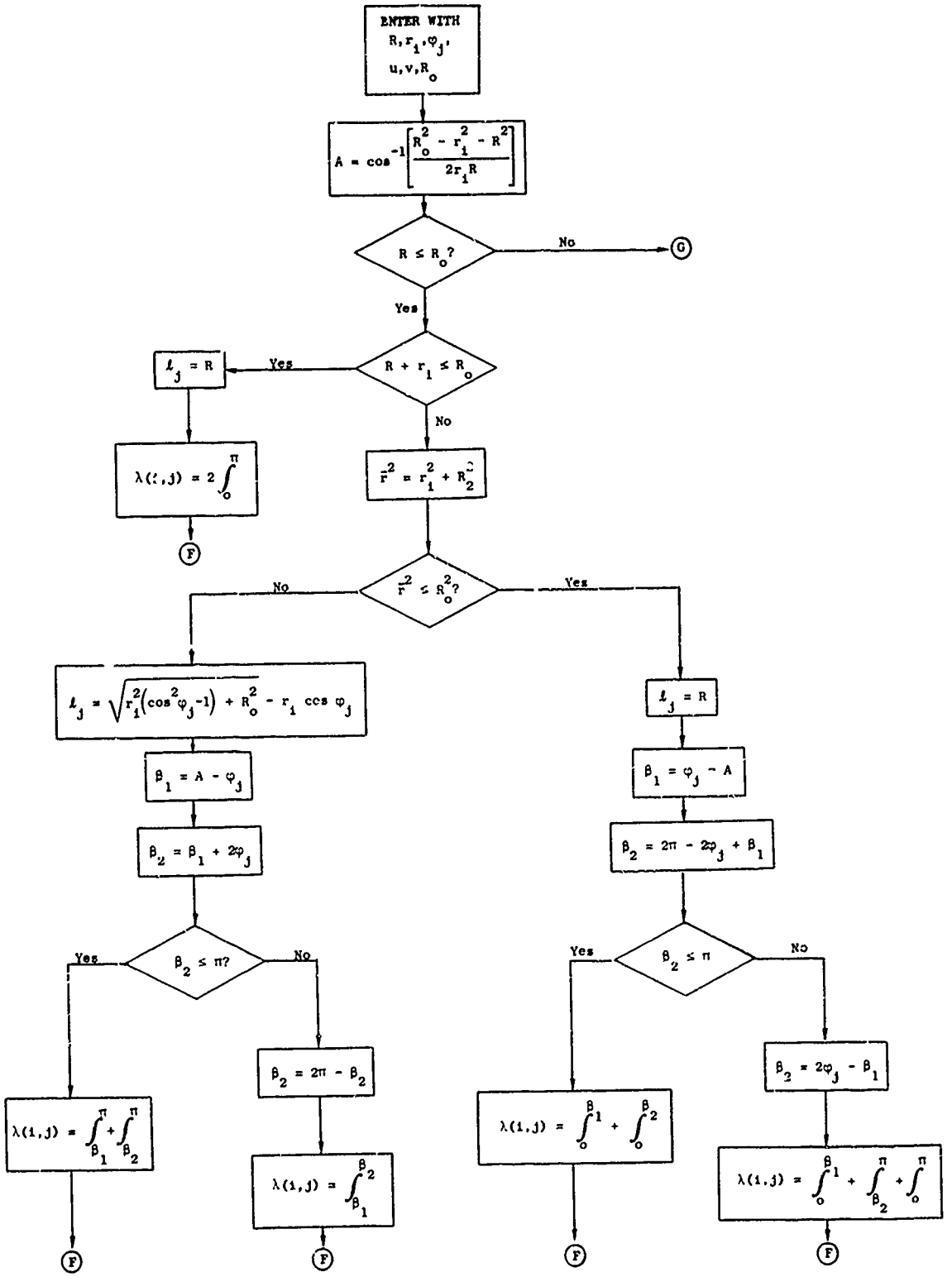


FIGURE 3.9 DETERMINATION OF CONDITIONAL DETECTION RATE

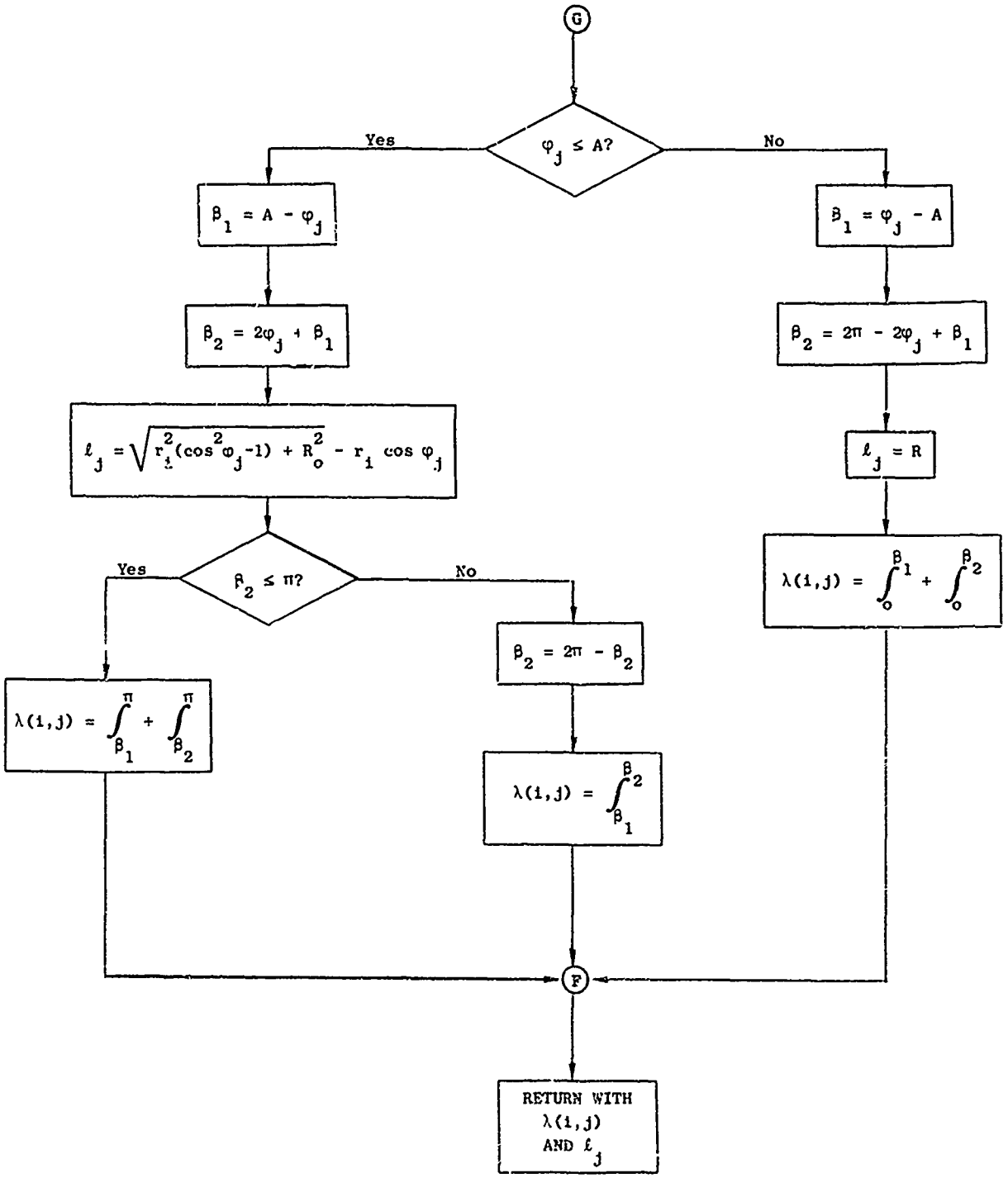


FIGURE 3.9 DETERMINATION OF CONDITIONAL DETECTION RATE (Concluded)

$$\lambda = n_T \sum_i \sum_j \lambda'(i,j) p(r_j) p(\varphi_j)$$

where $\lambda'(i, j)$ is obtained from $\lambda(i, j)$ with $n_T = 1$. In terms of earlier notation we have then

$$\beta_H = n_H \lambda_H$$

$$\beta_D = n_D \lambda_D$$

$$\lambda_c = n_H n_D \lambda_{HD} ,$$

where λ_H, λ_D, and λ_{HD} are, respectively, the rate of submarine encounter of a single HVS, the rate of submarine encounter of a single decoy, and the rate of single HVS encounter with a single decoy. Components of $p(\varphi_j)$ are also included in Figure 3.9. We take $p(\varphi_j)$ as

$$p(\varphi_j) = \frac{\ell_j}{\sum_j \ell_j} ,$$

where ℓ_j is the distance from the submarine along its heading φ_j to the closer of two boundaries: (1) the detection circle R centered at the submarine, and (2) the boundary of the operating area A_o. Thus, $p(\varphi_j)$ is an ad hoc measure of the likelihood of the submarine's being on a particular heading φ_j. Basically, this measure discounts the probability of the submarine's being on a heading directed out of A_o when he is close to the boundary. The probability $p(r_i)$ is determined as before, i.e., $p(r_i) = w(r_i)$.

Conditioning on r_i and φ_j and computing λ as a weighted average were introduced to account for the facts that: (1) we are dealing with a bounded area, and (2) when the submarine is in the search process and has no contacts the target density in the unswept area is affected by the size of the swept area. It is interesting to compare the weighted average λ with the unweighted formulation, call it λ^*, where

$$\lambda^* = \frac{n_T R}{A_o \pi} \; 2\int_o^{\pi} f(\beta;u,v)d\beta \quad .$$

λ^* is the formulation used in the earlier work (Ref. 3). A comparison between λ and λ^* is shown in Figure 3.10. The deviation between the two is most extreme for larger values of detection range R, as would be expected. Deviation also increases with target speed. An interesting comparison between λ and λ^* is in the computation of

$$p_{12} = \frac{n_D \lambda_H}{n_H \lambda_H + n_D \lambda_D} \quad .$$

Consider as an example $n_H = 1$, $R_H = 50$, $n_D = 1$ and 10, and $R_D = 100$ and 200. Table 3.1 shows the comparison.

Table 3.1

COMPARISON OF DETECTION RATE FORMULATIONS

	$R_D = 100$	$R_D = 200$
$n_D = 1$	.6689 .6691	.8011 .8650
$n_D = 10$	.9528 .9528	.975 .984

Table Entry $\begin{cases} p_{12}(\lambda^*) \\ p_{12}(\lambda) \end{cases}$

The differences indicated by this table range from small to nonexistent. It seems appropriate that in future work a simple approximation to λ based on Figure 3.8 should be adequate for most situations.

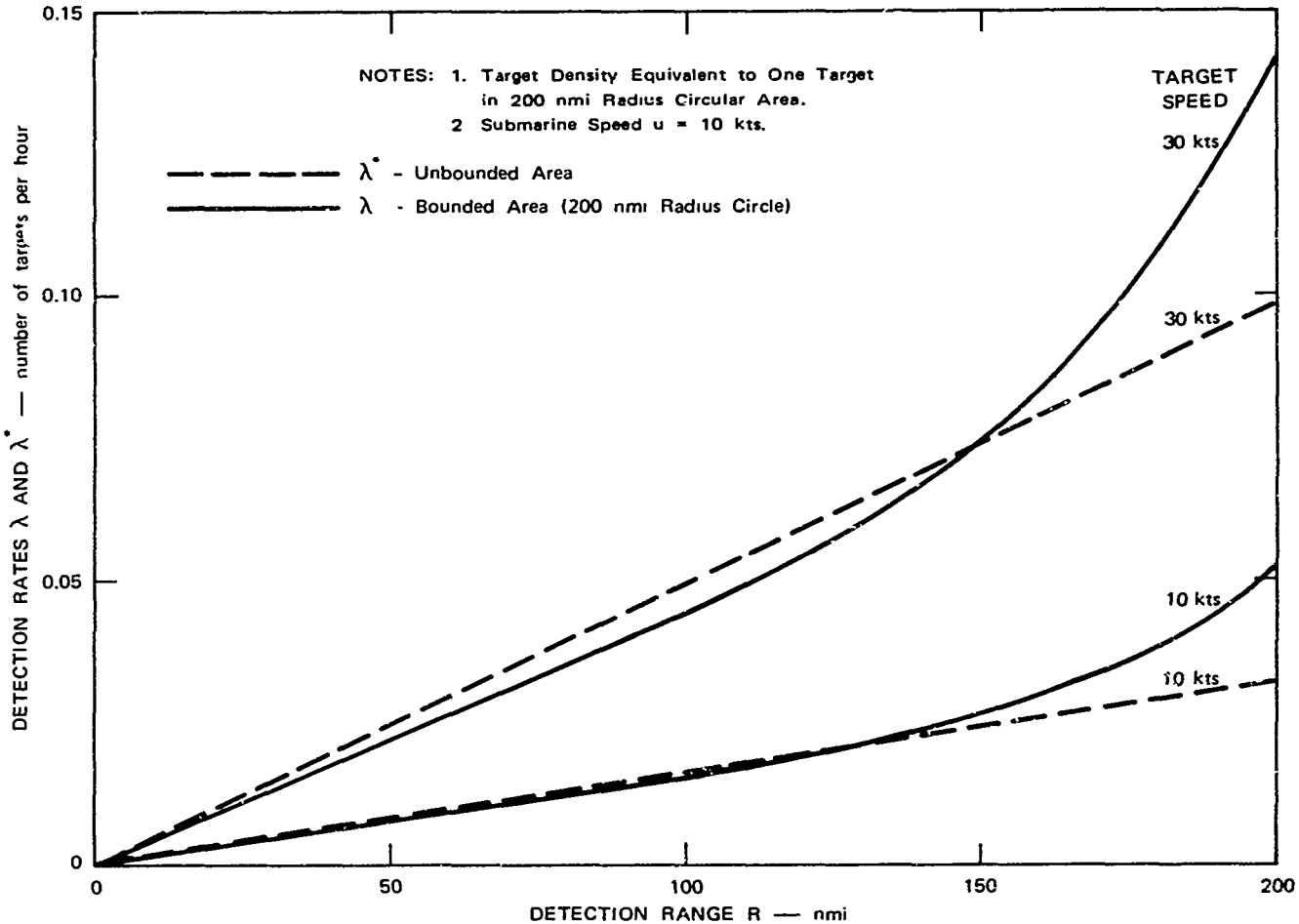


FIGURE 3.10 DETECTION RATE VERSUS DETECTION RANGE

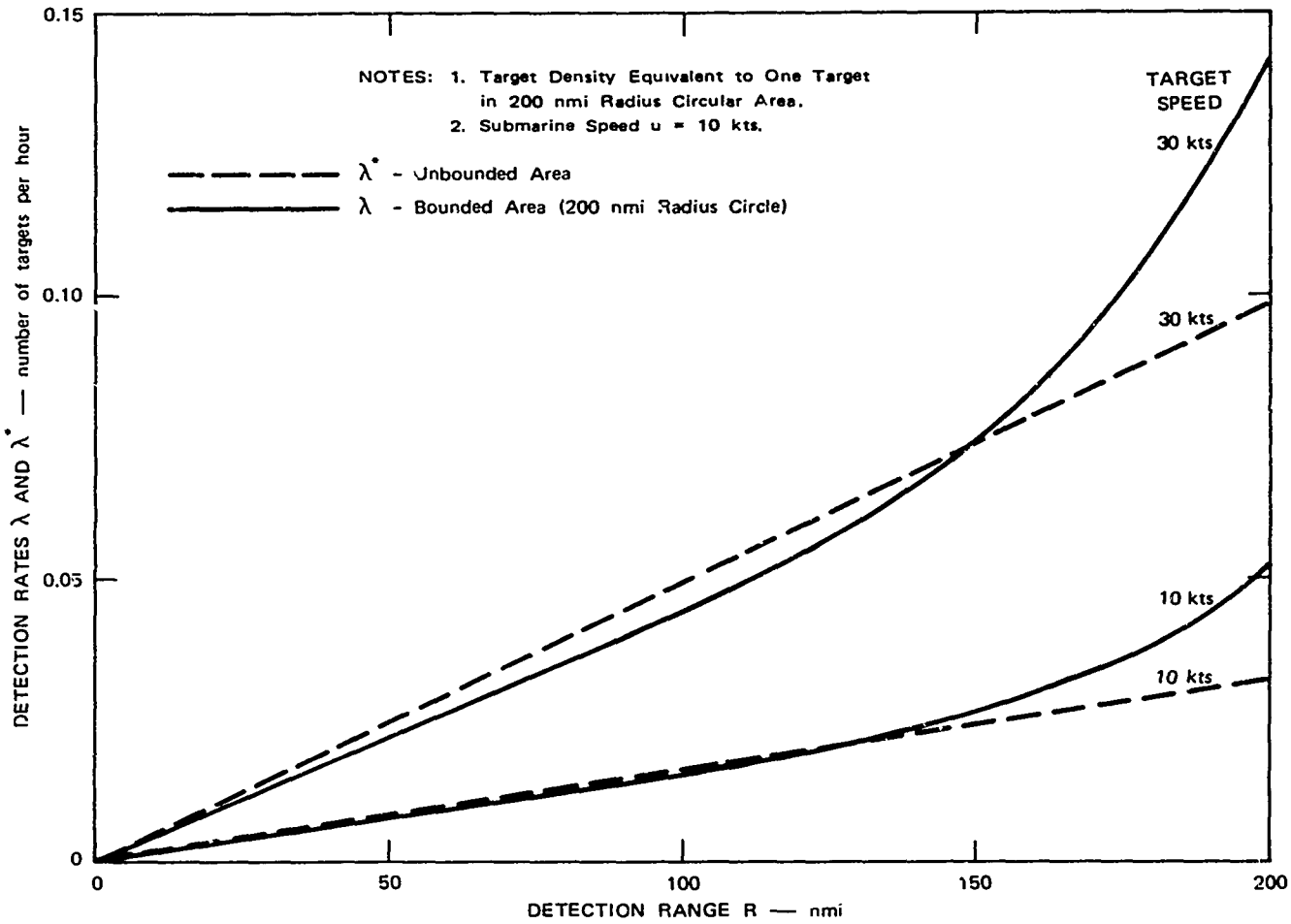


FIGURE 3.10 DETECTION RATE VERSUS DETECTION RANGE

4. MODEL SOLUTION AND COMPUTATION

Given the measure of effectiveness (MOE) and the details of model structure, it remains to show how the MOE is related to the model and how it is computed. Recall that the MOE is expressed in terms of the function $\varphi_{ij}(t)$, the probability that at time t the process is in State j given that at time zero it was in State i. Assuming that at time zero the process starts in the entry state, we easily narrow the scope to $\varphi_{ij}(t)$. Further, in the six-state model we are interested in misclassification of a decoy because the submarine is removed as a threat so here we focus on $\varphi_{19}(t)$ and $\varphi_{110}(t)$.

Since the six- and ten-state models entail different assumptions, they have different methods of solution. First, the six-state model is treated and then the ten-state.

4.1 The Six-State Model

The six-state model is described by a transition probability matrix P and a holding time pdf matrix H(t) where

$$H(t) = [h_{ij}(t)] \quad \text{and} \quad h_{ij}(t) = h_i(t) \quad \text{all } j \quad .$$

Recall that the superscript "e" denotes exponential transform, i.e.,

$$h_i^e(s) = \int_0^\infty d\tau\, e^{-st} h_i(t)dt \quad .$$

Howard (Ref. 4) discusses process flow graph methods for obtaining $\varphi_{13}(t)$ from P and H(t). The flow graph is modified by removing the self-loop on node 3 (the trapping state). Howard shows that the transmission $\overline{t}_{13}(s)$

from node 1 to the output node of the modified flow graph is $s\omega^{e}_{13}(s)$. Hence, $\varphi^{e}_{13}(s) = s^{-1}\vec{t}_{13}(s)$. This transform turns out to be a ratio of polynomials which can easily be inverted by finding the roots of the denominator. The modified process flow graph is shown in Figure 4.1.

The transmission from node i to j of a flow graph is defined as

$$\vec{t}_{ij} = \frac{1}{\Delta} \sum_{k \epsilon P_{ij}} \vec{t}^{k}_{ij} \Delta^{k}_{ij} \quad ,$$

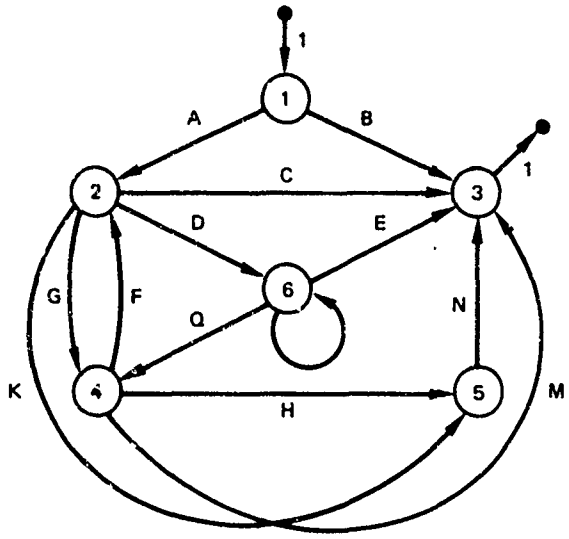
where Δ is one plus the sum of all loop products of all loops in the graph, P_{ij} is the set of paths leading from i to j, $\vec{t}^{k}_{ij}$ is the path transmission of the k^{th} path leading from i to j, and Δ^{k}_{ij} is equal to one plus the sum of the loop products of all loops that share no node with the k^{th} path from i to j. In our case $\vec{t}_{ij}$ is a function of s so we write $\vec{t}_{ij}(s)$. The simple loop products for the graph in Figure 4.1 are: -P, -GF, -GHK, -DFQ, -DHKQ. They are negative by definition. The multiple loop products are obtained by considering all possible combinations of simple loops that have no nodes in common and multiplying their products. The multiple loop products for Figure 4.1 are: FGP, GHKP. Hence, we have

$$\Delta = 1 - [P + GF + GHK + DFG + DHKQ] + [FGP + GHKP] \quad .$$

There are seven paths from node 1 to node 3. They can be identified by their transmissions $\vec{t}^{k}_{13}$ = B, AC, ADE, AGM, AGHN, ADQM, ADQHN. The sums Δ^{k}_{13} are one for all paths except AC, AGM, and AGHN for which they are 1-P and B for which it is Δ. Hence,

$$\vec{t}_{13}(s) = \frac{1}{\Delta}[B\Delta + AC(1-P) + ADE + AGM(1-P) + AGHN(1-P) + ADQM + ADHQN]$$

$$= B + \frac{A}{\Delta}[DE + (C + GM + GHN)(1-P) + DQ(M+HN)] \quad .$$



$A = p_{12}h_1^*(s)$ $\quad H = p_{45}h_4^*(s)$

$B = p_{13}h_1^*(s)$ $\quad K = p_{52}h_5^*(s)$

$C = p_{23}h_2^*(s)$ $\quad M = p_{43}h_4^*(s)$

$D = p_{26}h_2^*(s)$ $\quad N = p_{53}h_5^*(s)$

$E = p_{63}h_6^*(s)$ $\quad P = p_{66}h_6^*(s)$

$F = p_{42}h_4^*(s)$ $\quad Q = p_{64}h_6^*(s)$

$G = p_{24}h_2^*(s)$

FIGURE 4.1 MODIFIED PROCESS FLOW GRAPH FOR SIX STATE MODEL

Each of the letters in Δ and $\vec{t}_{13}(s)$ is identified with some $p_{ij}h_i^e(s)$. To obtain $\vec{t}_{13}$ as a function of s we must specify the forms of $h_i^e(s)$. For $i = 1, 4, 5$, $h_i(t)$ is of the form

$$h_i(t) = \alpha_i e^{-\alpha_i t} \quad , \quad t \geq 0$$

which implies

$$h_i^e(s) = \frac{\alpha_i}{s + \alpha_i} \quad .$$

For $i = 2$ and 6, $h_i(t)$ is of the form

$$h_i(t) = \frac{(\alpha_i n_i)^{n_i}}{(n_i-1)!} t^{n_i-1} e^{-n_i\alpha_i t} \quad , \quad t \geq 0$$

which implies

$$h_i^e(s) = \left[\frac{n_i\alpha_i}{s + n_i\alpha_i}\right]^{n_i} \quad .$$

To simplify algebra and subscripting the following identifications are made

$$\alpha_1 = a = n_H\lambda_H + n_D\lambda_D$$

$$\alpha_2 = b = n\mu_D$$

$$\alpha_4 = d = n_H\lambda_H + (n_D-1)\lambda_D$$

$$\alpha_5 = f = n_H\lambda_H + n_D\lambda_D$$

$$\alpha_6 = c = n\mu_D \quad .$$

Now $\varphi_{13}^e(s) = s^{-1}\tilde{t}_{13}(s)$ can be written in terms of a ratio $P(s)/Q(s)$ of polynomials $P(s)$ and $Q(s)$, specifically,

$$\varphi_{13}^e(s) = \frac{B}{s} + \frac{A}{s}\frac{P(s)}{Q(s)}$$

where

$$P(s) = \sum_{i=1}^{N_P(n)} p_i s^{i-1}$$

$$Q(s) = \sum_{i=1}^{N_Q(n)} q_i s^{i-1} \quad .$$

In the case $n = 1$, we find that $N_P(n) = 4$, $N_Q(n) = 5$, and

$$p_4 = bp_{23}$$

$$p_3 = bcp_{25}p_{63} + b(d+f)p_{23} + bdp_{24}p_{43}$$

$$p_2 = bc(d+f)[p_{26}p_{63} + (1-p_{66})p_{23}]$$
$$+ bcd[p_{24}p_{43}(1-p_{66}) + p_{26}p_{64}p_{43}]$$
$$+ bdf(p_{23}+p_{24}p_{43}+p_{24}p_{45}p_{53})$$

$$p_1 = bcdf[p_{26}p_{63} + p_{26}p_{64}(p_{43}+p_{45}p_{53})$$
$$+ (1-p_{66})(p_{23}+p_{24}p_{43}+p_{24}p_{45}p_{53})]$$

and

$$q_5 = 1$$

$$q_4 = b + c(1-p_{66}) + d + f$$

$$q_3 = (b+d+f)c(1-p_{66}) + bd(1-p_{24}p_{42}) + (b+d)f$$

$$q_2 = bcd(1-p_{66}-p_{24}p_{42}+p_{24}p_{42}p_{66}-p_{26}p_{42}p_{64})$$
$$+ (b+d)cf(1-cp_{66})$$
$$+ bdf(1-p_{24}p_{42}-p_{24}p_{45}p_{52})$$

$$q_1 = bcdf(1-p_{66}-p_{24}p_{42}-p_{24}p_{45}p_{52}$$
$$- p_{26}p_{42}p_{64}-p_{45}p_{52}p_{26}p_{64}+p_{24}p_{42}p_{66}$$
$$+ p_{24}p_{45}p_{52}p_{66}) \quad .$$

For cases with $n > 1$ the algebra involved in obtaining P(s) and Q(s) is extremely cumbersome. For this reason a computer program utilizing the IBM FORMAC algebraic language has been developed to handle the algebraic

manipulations. FORMAC is an extension of PL/1 and provides for the symbolic manipulation of mathematical expressions, e.g., the expression SIN(X) can be differentiated, resulting in the expression COS(X). Expressions can be differentiated, evaluated, replaced, compared, and parsed. The language is explained in detail in Ref. 7 and the program developed for this project is listed in Appendix D. The results of the program in terms of P(s) and Q(s) for n = 2, 3, and 10 are listed in Appendix E.

Returning to the expression for $\varphi^e_{13}(s)$,

$$\varphi^e_{13}(s) = \frac{B}{s} + \frac{A}{s}\frac{P(s)}{Q(s)}$$

and substituting for A and B, we obtain

$$\varphi^e_{13}(s) = p_{13}\frac{1}{s(s+a)} + p_{12}a\frac{P(s)}{s(s+a)Q(s)} \quad .$$

The problem now is to invert the exponential transforms. For the first term on the right we have simply from a table of transforms

$$p_{13}\frac{1}{s(s+a)} \longrightarrow p_{13}(1 - e^{-at}) \quad .$$

The second term on the right is more involved. Let

$$T(s) = s(s+a)Q(s)$$

$$W(s) = P(s)/T(s) \quad .$$

We can numerically find the roots of Q(s) since we know the coefficients; call these roots $r_1, \ldots, r_N$, some of which may be complex. These are also roots of T(s). T(s) in addition has roots $r_{N+1} = -a$ and $r_{N+2} = 0$. Now it is known (e.g., Ref. 8) from algebra that W(s) can be decomposed into partial fractions

$$W(s) = \sum_{k=1}^{N+2} \frac{Z_k}{s - r_k} \quad ,$$

where

$$Z_k = P(r_k)/T'(r_k)$$

$$= \frac{P(r_k)}{r_k(r_k+a)Q'(r_k) + r_k Q(r_k) + (r_k+a)Q(r_k)} \quad .$$

Then inverting W(s) we get

$$w(t) = \sum_{k=1}^{N+2} Z_k e^{r_k t} \quad .$$

So

$$p_{12}{}^a \frac{P(s)}{s(s+a)Q(s)} \longrightarrow p_{12}{}^a \sum_{k=1}^{N+2} Z_k e^{r_k t} \quad .$$

Putting all the terms together we have for the six-state model

$$\varphi_{13}(t) = p_{13}(1 - e^{-at}) + p_{12}{}^a \sum_{k=1}^{N+2} Z_k e^{r_k t} \quad .$$

4.2 The Ten-State Model

The ten-state model is restricted to exponential type pdf's, so more efficient methods of solution are available for it. The exponential pdf's insure that the ten-state model is a Markov process. We are interested in $\varphi_{19}(t)$ and $\varphi_{110}(t)$, or equivalently $\varphi_9(t)$ and $\varphi_{10}(t)$ in the vector $\varphi(t)$ defined by

$$\varphi(t) = \varphi(0)\Phi(t) \quad ,$$

where $\Phi(t) = [\varphi_{ij}(t)]$ and $\varphi(0) = (1, 0, 0, \ldots, 0)$. Howard (Ref. 4) shows that $\Phi(t) = e^{At}$ (matrix exponential) where $A = \Lambda(P - I)$ with Λ a diagonal matrix with elements corresponding to the rates of the exponential holding times. Specifically,

$$\Lambda = \text{diag}(\lambda_i)$$

where

$$\lambda_1 = \beta_H + \beta_D \qquad \lambda_6 = n_H\lambda_H + (n_D-1)\lambda_D + \lambda_R$$

$$\lambda_2 = \mu_D \qquad \lambda_7 = (n_H-1)\lambda_H + n_D\lambda_D + \lambda_R$$

$$\lambda_3 = \mu_H \qquad \lambda_8 = n_H\lambda_H + n_D\lambda_D$$

$$\lambda_4 = \mu_D \qquad \lambda_9 = 0$$

$$\lambda_5 = \mu_H \qquad \lambda_{10} = 0 \quad .$$

An iterative scheme is used for calculating the vector $\varphi(t)$ Given $\varphi(t) = \varphi(0)e^{At}$ we can write

$$\varphi(t+\Delta t) = \varphi(0)e^{A(t+\Delta t)} = \varphi(0)e^{At}e^{A\Delta t} = \varphi(t)e^{A\cdot\Delta t} \quad .$$

Thus, to compute $\varphi(t)$ at points t that are Δt apart we need only compute one matrix exponential, $e^{A\Delta t}$. To compute the matrix exponential e^{Z} we use the Pade approximation

$$e^{Z} = Z_1^{-1}Z_2 \quad ,$$

where

$$Z_1 = 1680I - 840Z + 180Z^2 - 20Z^3 + Z^4$$

$$Z_2 = 1680I + 840Z + 180Z^2 + 20Z^3 + Z^4 \quad .$$

This expression for e^Z is an extension of a table of Pade approximations for the exponential function contained in Ref. 9.

5. LOGICAL FLOW

This section summarizes the preceding sections on model structure and geometrical analysis in terms of logical flow charts. Figure 5.1 gives a summary flow chart for the six-state model and Figure 5.2 for the ten-state model. These charts show the relationships among the inputs required, preliminary calculations, the major loop on number of decoys, transition probability and rate determination, partial fraction expansion or matrix exponentiation, and calculation of trapping probabilities.

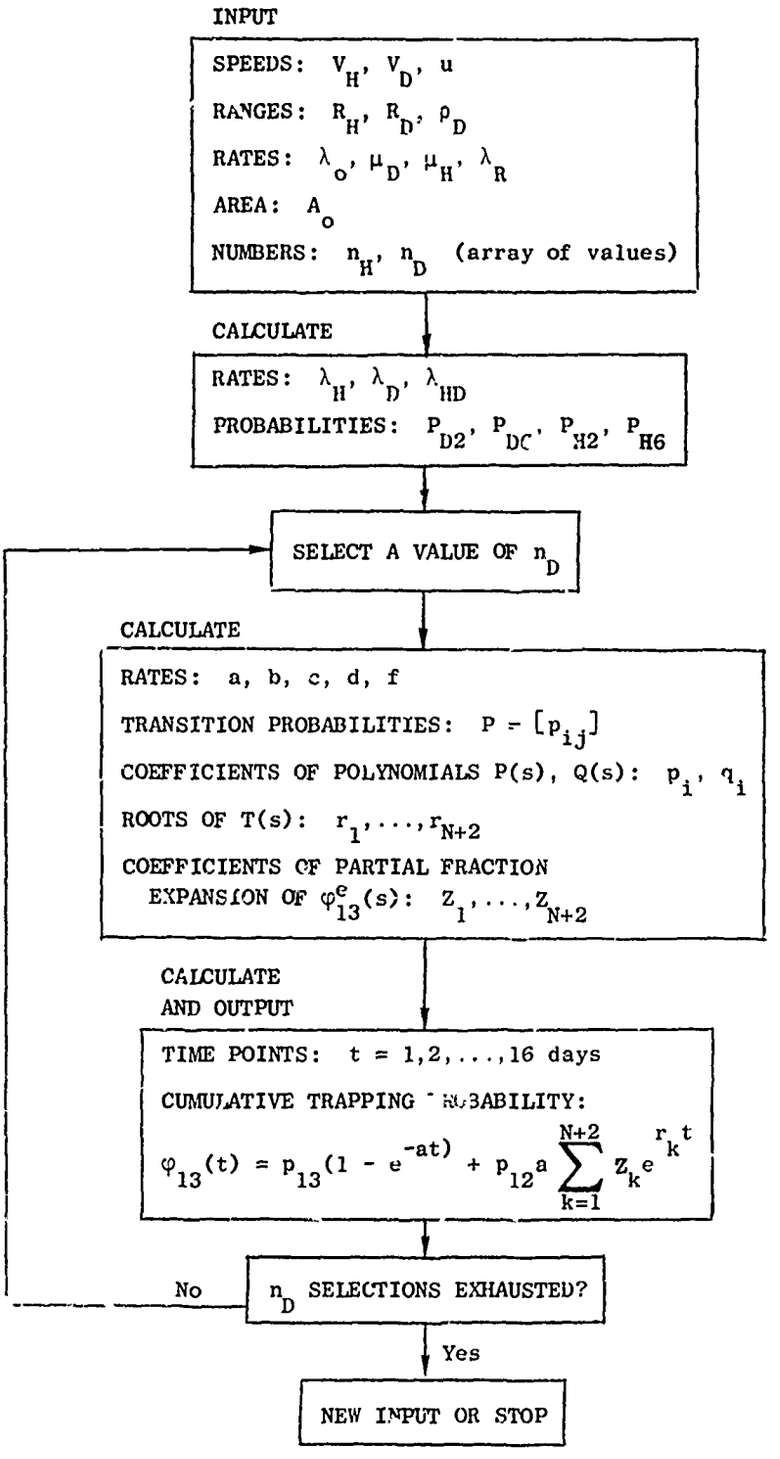


FIGURE 5.1 SUMMARY LOGICAL FLOW OF SIX-STATE MODEL

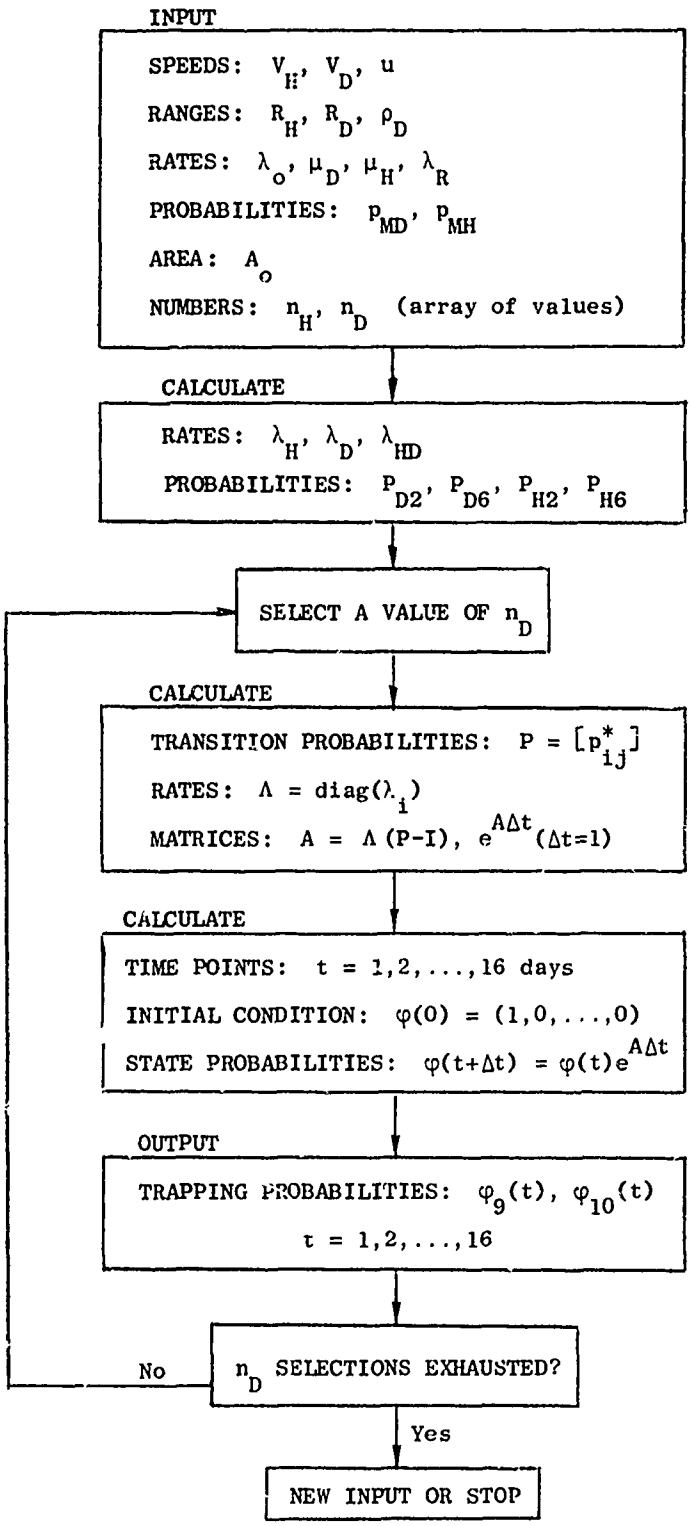


FIGURE 5.2 SUMMARY LOGICAL FLOW OF TEN-STATE MODEL

6. MODEL COMPARISON AND SENSITIVITY

This section compares the old and the new models and presents some sensitivity analysis. The purpose is to evaluate what has been developed in order to aid in its use and future modification.

6.1 Old and New Model Geometry

The development of the original three-state semi-Markov model into the model used in this study involves incorporation of the problem geometry. There are two modifications: (1) representation of the boundary conditions imposed by the finite circular operating area, and (2) representation of the dissipation of swept area during the classification of contacts. This section assesses the significance of these modifications.

The effect of the boundary representation is examined by comparing results for a single-sub/single-HVS search with no decoys.* To give a broad basis to the comparison the detection radius is varied parametrically. The difference between the old model and the new is barely perceptible; the comparison is shown in Tables 6.1 and 6.2. The first table shows probability of encounter for a 1-day period, while the second table shows it for a 7-day period. The maximum difference shown is 0.03. This difference is quite small, insignificant in the case of a single target. However, the difference might be magnified by the inclusion of decoys.

*The other parameter values in this case include submarine speed, 10 knots; HVS speed, 15 knots; and a 200-nmi radius operating area.

Table 6.1

BOUNDARY REPRESENTATION COMPARISON
(One Day)

HVS Detection Radius	Probability of Encounter (One Day)	
	Old Model	New Model
10	0.06	0.06
20	0.12	0.11
30	0.17	0.17
40	0.23	0.21
50	0.27	0.26
60	0.32	0.29
70	0.36	0.34
80	0.40	0.37
90	0.44	0.41
100	0.47	0.45
110	0.50	0.49
120	0.54	0.52

Table 6.2

BOUNDARY REPRESENTATION COMPARISON
(Seven Days)

HVS Detection Radius (nmi)	Probability of Encounter (Seven Days)	
	Old Model	New Model
10	0.36	0.36
20	0.59	0.57
30	0.74	0.72
40	0.83	0.81
50	0.89	0.88
60	0.93	0.91
70	0.96	0.94
80	0.97	0.96
90	0.98	0.98
100	0.99	0.98
110	0.99	0.99
120	0.99	0.99

Unfortunately, with the inclusion of decoys the boundary representation becomes confounded with the representation of dissipation of swept area.

Figure 6.1 shows two cases of model comparison involving decoys; the variable parameter is decoy detection range. Submarine speed is 10 knots; the other parameter values are shown below. Case 1 is a single CV in a nonconvergence zone with a larger number of lower fidelity devices; Case 2 is an integrated CTG in a convergence zone with a smaller number of higher fidelity devices. In this comparison there is considerable difference between the models. The maximum difference in Case 1 is 0.20 (difference between 0.19 and 0.39); the maximum difference in Case 2 is 0.25 (difference between 0.32 and 0.57). It will be shown below that the difference is due to the zero speed of the decoys, which implies a low rate of dissipation of swept area.

Case 1	Case 2
CV in NCZ	CTG in CZ
V_D = 0 kt	V_D = 0 kt
n_D = 12	n_D = 5
T_c = 3 hr	T_c = 10 hr
R_H = 28 nmi	R_H = 90 nmi
V_H = 15 kt	V_H = 15 kt

The vertical broken lines in Figure 6.1 indicate decoy ranges which are subject to sensitivity analysis on number of decoys (Case 2 only) in Figure 6.2. The value R_D = 70 nmi is the break point between the two models in Case 2 of Figure 6.1, and the difference between models

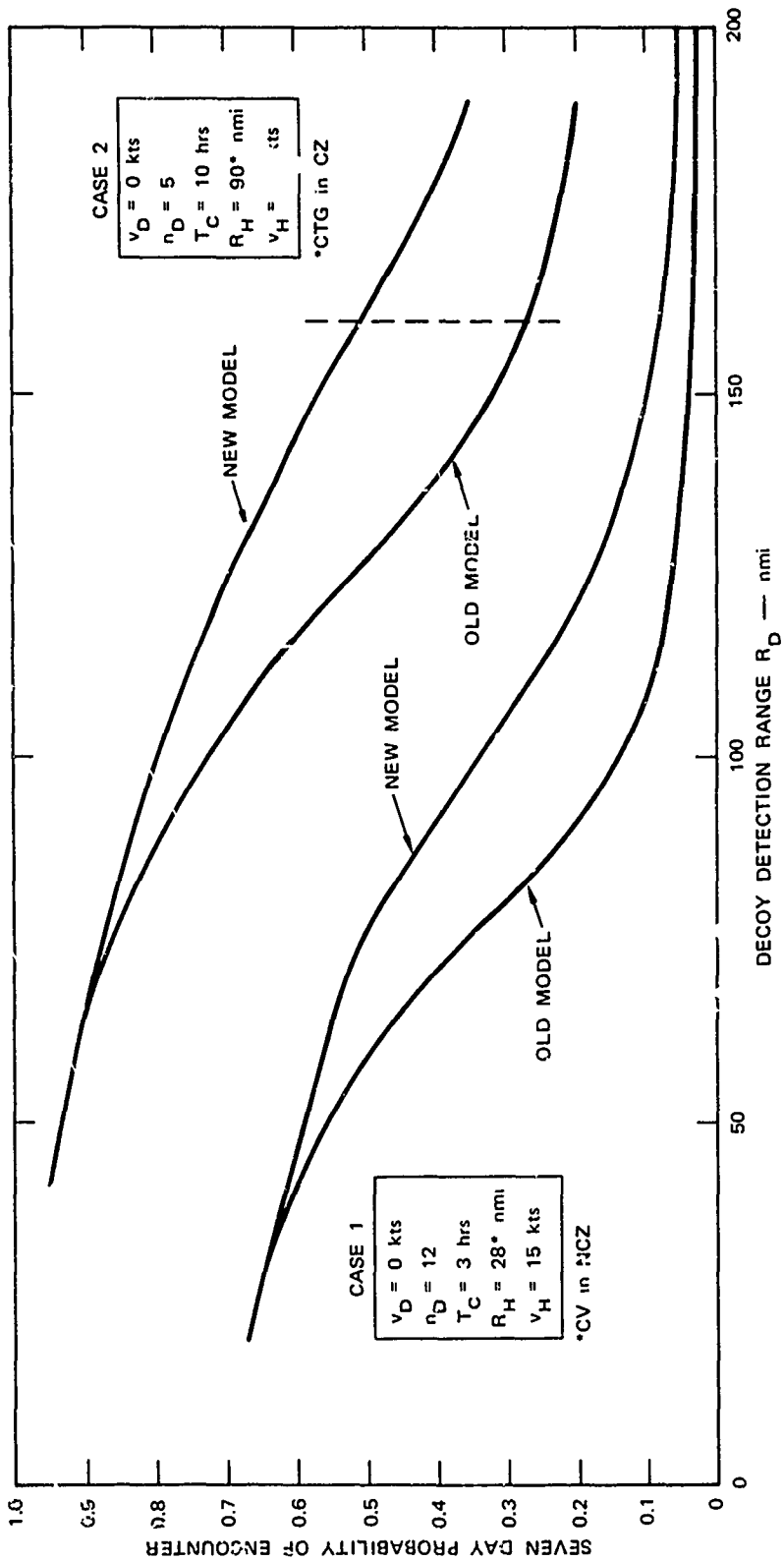


FIGURE 6.1 MODEL GEOMETRY COMPARISON AS A FUNCTION OF DECOY DETECTION RANGE (CASES 1 AND 2)

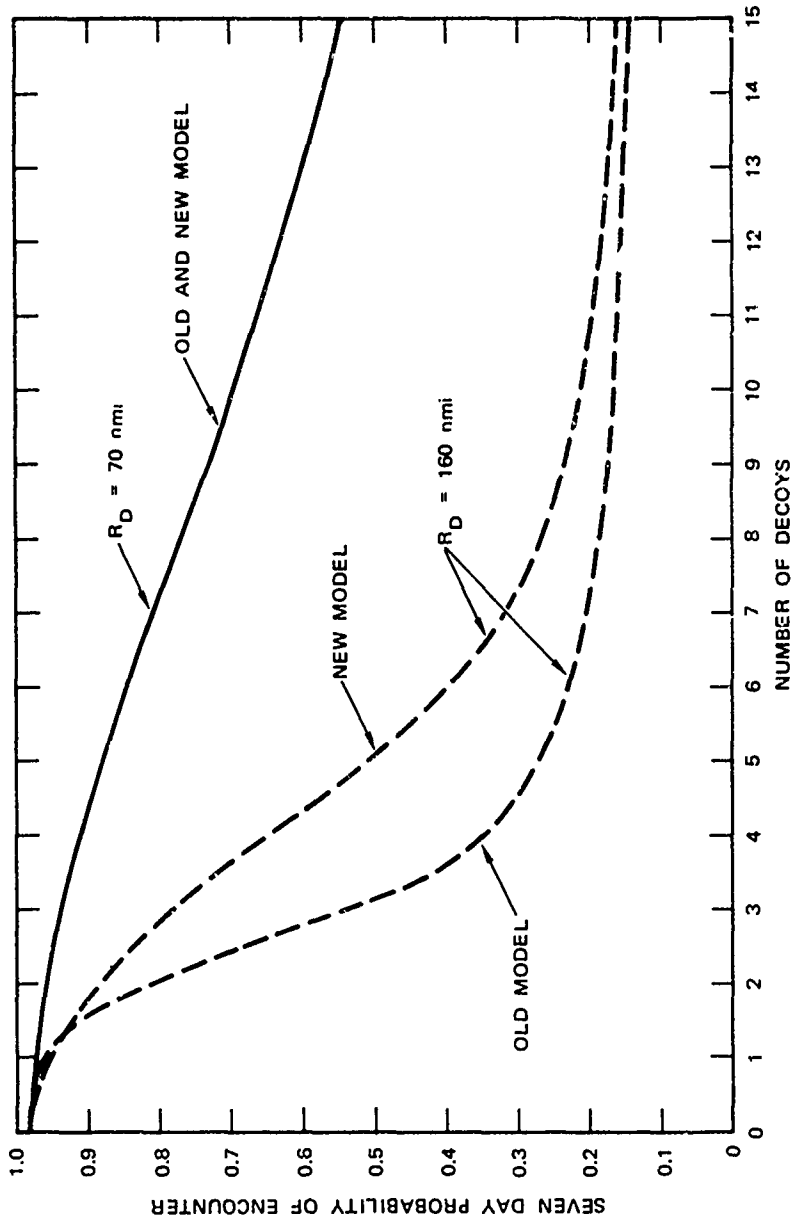


FIGURE 6.2 MODEL GEOMETRY COMPARISON AS FUNCTION OF NUMBER OF DECOYS (CASE 2)

is insensitive to decoy numbers in Figure 6.2. On the other hand, R_D = 160 nmi is a point of wide disparity in Figure 6.1, and Figure 6.2 indicates this disparity is a function of number of decoys, with disparity maximum at four decoys and decreasing with fewer or more decoys.

Thus far it is apparent that geometry representation can have a large effect, but the mechanism of the effect is not clear. Case 3 shown in Figure 6.3 clarifies this mystery; Case 3 is a high speed CV in a convergence zone. The important fact illustrated in this figure is that the large difference evident between the new and old models at a decoy speed of zero collapses when the decoy speed is increased to 10 knots.

The conclusions are: (1) geometry representation can have an appreciable effect, (2) the effect depends on the parameters of the problem (e.g., decoy detection range, speed, capture time, and number), and (3) the effect is due primarily to the representation of the dissipation of swept area.

6.2 Probability of Misclassification and Information Retention Time

Inclusion of misclassification probability and retention by the submarine of information on the last decoy classified are the other two major modifications in addition to geometry. Figure 6.4 illustrates the highly significant effect of the misclassification probabilities p_{MH} and p_{MD}. However, it has been found that retention time of information has virtually no effect over a wide range of situations for almost any reasonable retention time (e.g., 0 to 100 hours).

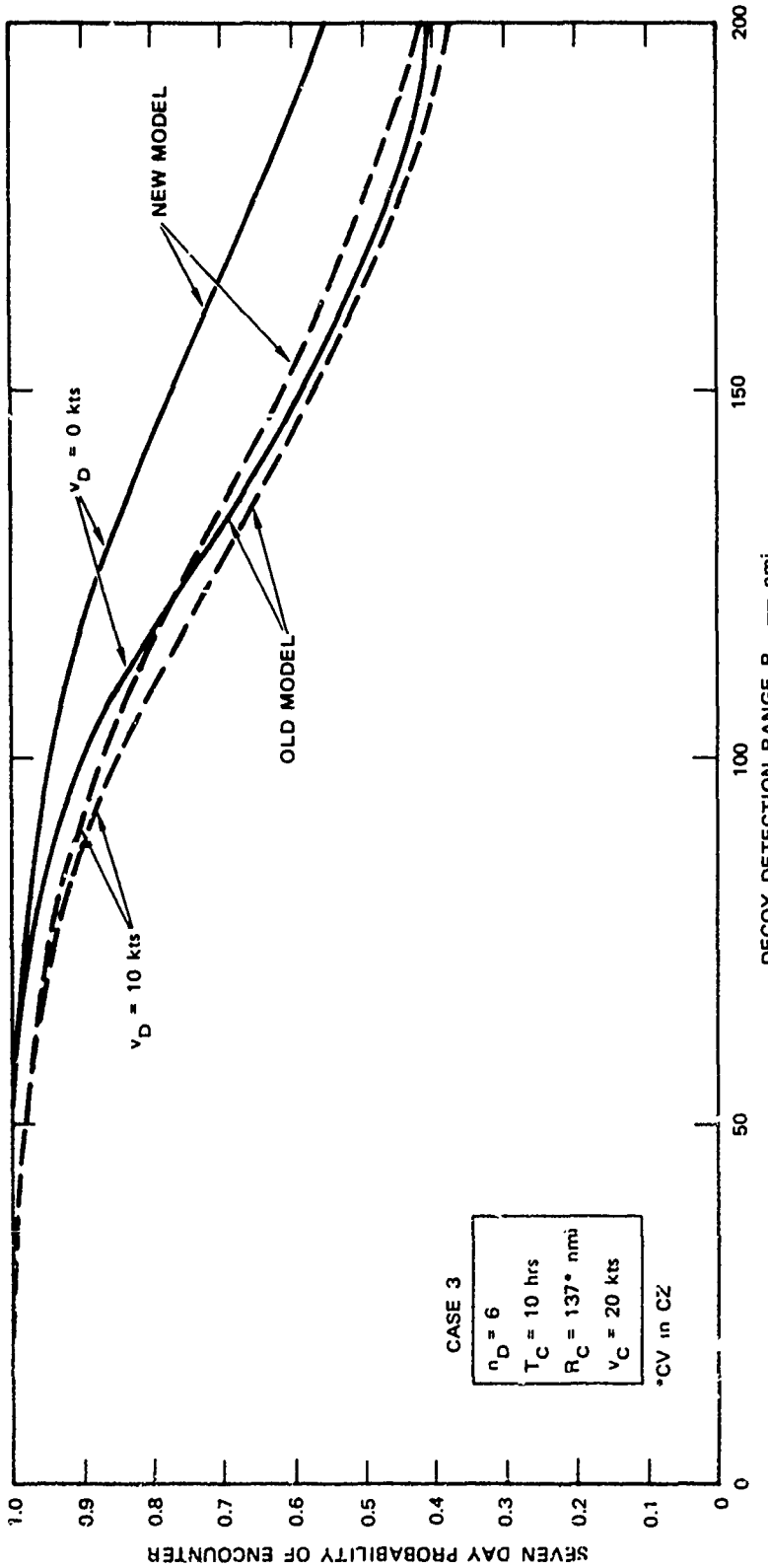


FIGURE 6.3 MODEL GEOMETRY COMPARISON AS A FUNCTION OF DECOY DETECTION RANGE (CASE 3)

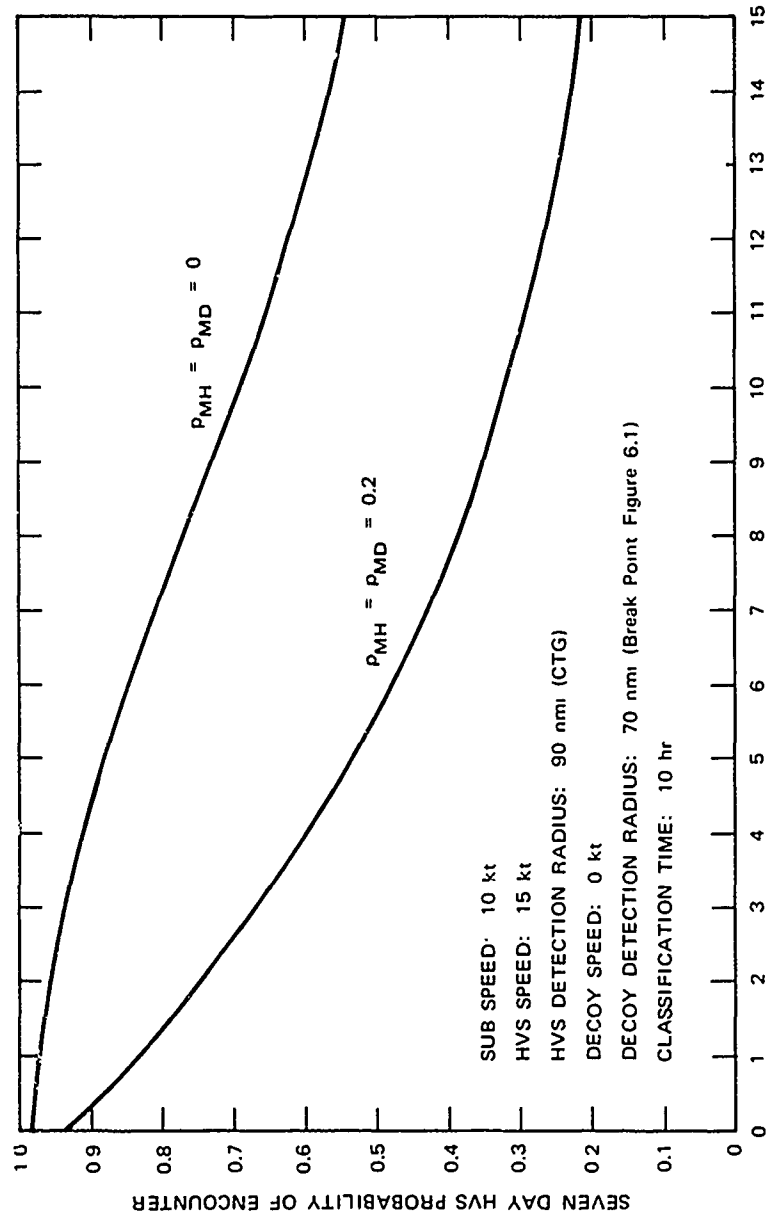


FIGURE 6.4 EFFECT OF MISCLASSIFICATION PROBABILITY

7. CONCLUSIONS AND RECOMMENDATIONS

The models discussed in this document are implemented as operational computer programs and have been used in concept evaluation studies reported elsewhere (Ref. 1). Experience in the development and use of these models allows some general conclusions to be drawn about the methodology.

Two conclusions are easy to draw: (1) potential for target misclassification is a highly significant part of the problem, and (2) retention of information on the last target classified is insignificant.

Another conclusion is that FORMAC is an interesting and powerful tool for analyzing continuous time semi-Markov process, but it can become cumbersome and time consuming in itself. Its use is best restricted to a small number of states (certainly no more than six), or to cases of very sparse transition matrices, where it can be very useful in deriving compact closed form solutions. However, for a larger number of states it appears a discrete time approximation is the better approach where consideration of non-Markovian processes is desirable. In the work described here it seems a 2-hour discrete time step would be the minimum necessary and perhaps longer, say 5 hours, would be acceptable.

Very significant is what has been discovered about the importance of geometry. Geometry representation can have an appreciable effect, but the effect is due primarily to the representation of the dissipation of swept area. The following simplified approach is suggested for future work:

(1) Ignore boundary conditions

(2) Adopt simple modification of unbounded area detection rates via approximations based on Figure 3.10 or an extension of this figure.

Appendix A

DISTRIBUTION OF TIME IN THE ENTRY STATE

The entry state is set up to allow consideration of nonstationary (time varying) encounter rates as the submarine enters the operating area. Let

$N_H(t)$ = encounter rate for HVSs at time t

$N_D(t)$ = encounter rate for decoys at time t

$\xi(t) = N_H(t) + N_D(t)$

$\eta_H(t) = \int_0^t N_H(\tau)\,d\tau$

$\eta_D(t) = \int_0^t N_D(\tau)\,d\tau$

$\eta(t) = \int_0^t \xi(\tau)\,d\tau$

T_D = time to detect an HVS with pdf $f_D(t)$

T_H = time to detect a decoy with pdf $f_H(t)$.

Assume

$$f_H(t) = N_H(t)e^{-\eta_H(t)} \qquad t \geq 0$$

$$f_D(t) = N_D(t)e^{-\eta_D(t)} \qquad t \geq 0 \quad .$$

If $N_H(t)$ and $N_D(t)$ are constants, then these are exponential distributions. The holding time pdf $h_1(t)$ is given by

pages 79, 80 blank.

$$h_1(t) = \xi(t)e^{-\eta(t)} \qquad t \geq 0 \quad .$$

It can then be shown that

$$P_{12} = P[T_D = \min(T_D, T_H)]$$

$$= \int_0^\infty \left[1 - e^{-\eta_D(t)}\right] N_H(t) e^{-\eta_H(t)} dt \quad .$$

To obtain $N_H(t)$ and $N_D(t)$ we use results from Ref. 6. Consider Figure A.1, where the submarine is heading directly in toward the center of the operating area. Assuming the detection circle is tangent to the

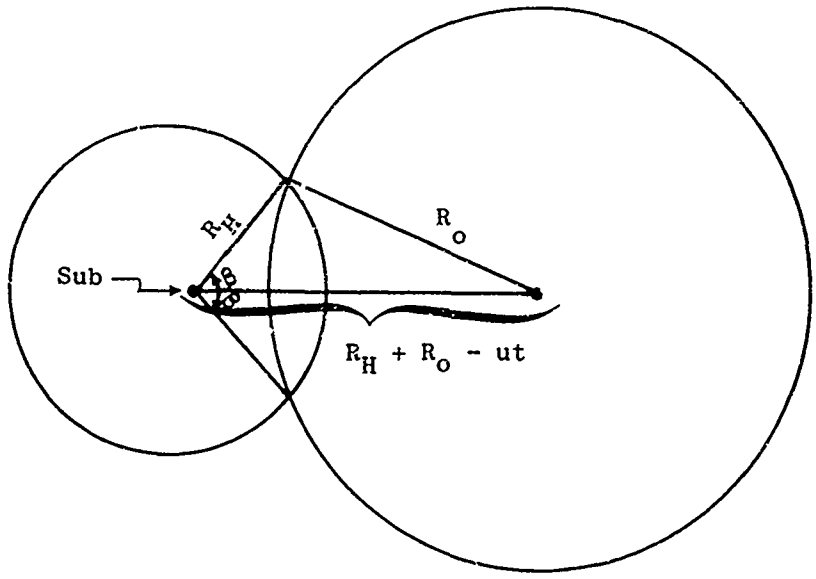


FIGURE A.1 ENTRY GEOMETRY FOR HVS AT TIME t

operating area at time zero, the distance between the submarine and the center of the operating area is $R_H + R_o - ut$ at time t, given the submarine speed u. The angle β is given by

$$\beta_H(t) = \arccos\left[\frac{R_H^2+(R_H+R_o-ut)^2-R_o^2}{2R_H(R_H+R_o-ut)}\right] .$$

Koopman (Ref. 6) shows the following. Assuming the submarine is operating in an infinite area, the encounter rate for HVSs experienced in the bearing region $(-\beta, \beta)$

$$\rho_H = \frac{2\delta_H R_H}{\pi}\int_o^{\beta} f(\beta; u, V_H)\,d\beta$$

where δ_H is target density. We assume this same formula applies approximately to the situation in Figure A.1, so that

$$N_H(t) = \frac{2n_H R_H}{\pi A_H}\int_o^{\beta_H(t)} f(\beta; u, V_H)\,d\beta$$

where A_H is A_o minus the overlap of the detection circle. A similar expression is used for $N_D(t)$. There are a number of logical exceptions to these expressions, but they are not discussed here; the objective of this appendix is to give only an overview of the approach examined. The next step is approximate $h_1(t)$. The approach taken is to fit

$$H_1(t) = \int_o^t h_1(\tau)\,d\tau$$

with an Erlang distribution

$$\hat{H}_1(t;\hat{\lambda},n) = 1 - e^{-n\hat{\lambda}t}\sum_{k=1}^{n}\frac{(\hat{\lambda}nt)^{k-1}}{(k-1)!} \quad , \quad t \geq 0$$

via a least squares technique on λ for fixed values of n, and then select the value of n with the least total squared deviation. In sixteen cases

examined, n = 1 occurred fifteen times, and n = 2 once. The n = 2 case is shown in Figure A.2. Since n = 1, the exponential distribution, is so heavily predominant and the procedure so time consuming, this special consideration of entry has been abandoned and entry is subsumed under search.

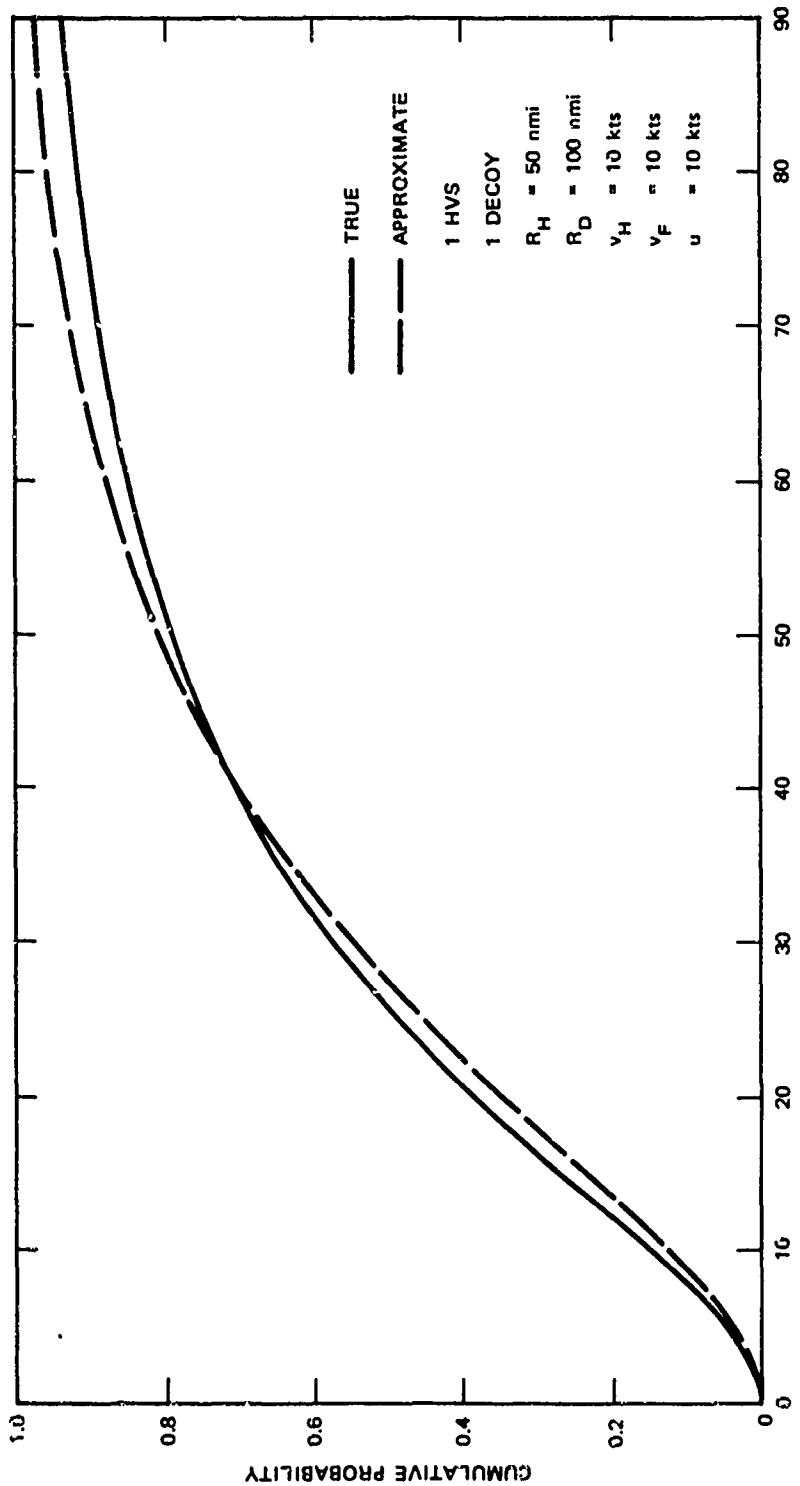


FIGURE A.2 ENTRY TIME DISTRIBUTION AND APPROXIMATION

Appendix B

THE ERLANG DISTRIBUTION

In this study the most general form of capture time distribution considered is the Erlang distribution. Let t be the random variable time, $n > 0$ a positive integer, and $\mu > 0$ a real number. The Erlang density and cumulative distribution functions are, respectively,

$$g(t;n,\mu) = \frac{(\mu n)^n}{(n-1)!} t^{n-1} e^{-n\mu t} \quad , \; t \geq 0$$

$$G(t;n,\mu) = 1 - e^{-n\mu t} \sum_{k=1}^{n} \frac{(\mu n t)^{k-1}}{(k-1)!} \quad , \; t \geq 0 \quad ,$$

with the convention $0! = 1$. The exponential transform is given by

$$g^e(s;n,\mu) = E[e^{-st}]$$

$$= \left[\frac{\mu n}{s+\mu n}\right]^n \quad .$$

The expectation, variance, and second moment are

$$E(t) = \frac{1}{\mu}$$

$$Var(t) = \frac{1}{n\mu^2}$$

$$E(t^2) = \frac{n+1}{n\mu^2} \quad .$$

pages 86, 87 and 88 Blank

By varying n and μ a large number of different shapes can be obtained, as illustrated in Figure B.1. In particular, when $n = 1$ the Erlang distribution is the exponential distribution, and when $n \to \infty$ then $Var(t) \to 0$, which implies constant capture time. Replacing μn by μ in the density and distribution functions yields the gamma distribution.

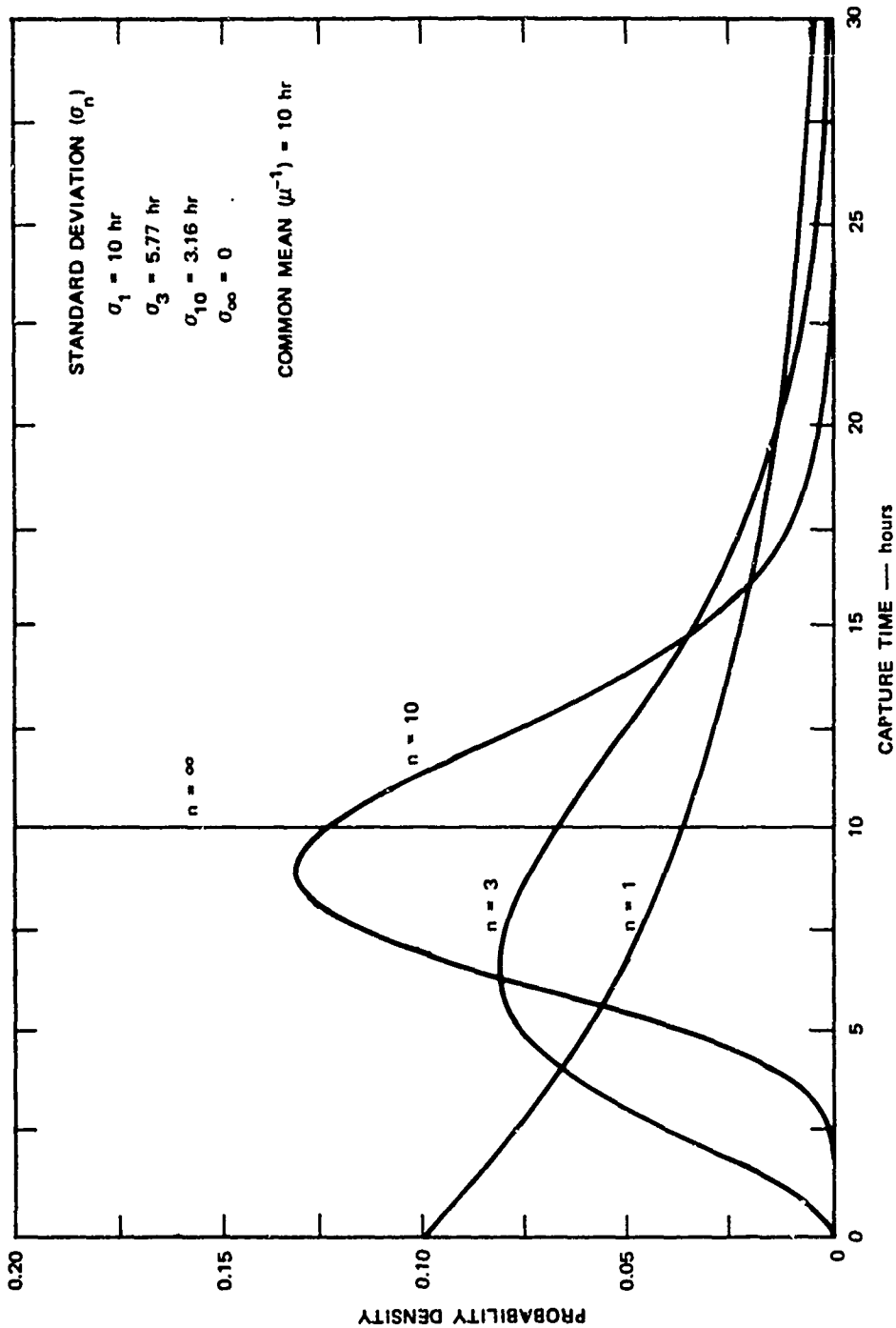


FIGURE B.1 DISTRIBUTION COMPARISON

Appendix C

DISTRIBUTION OF RANGE FROM CENTER OF OPERATING AREA

In obtaining expressions for $P(r_i|C_1)$ and $P(r_i|F_2)$, a pdf k(r) for the distance r from the center of the operating area A_o was used. This pdf has the form

$$k(r) = \frac{2\pi}{A_o} r \qquad 0 \le r \le R_o \quad ,$$

where R_o is the radius of A_o. The form follows from the assumption that position is uniformly distributed throughout A_o. Let (x, y) be the rectangular coordinates of position measured from the center of A_o. The assumption may be stated

$$f(x,y) = \begin{cases} \frac{1}{A_o} & \text{if } x^2 + y^2 \le R_o^2 \\ 0 & \text{otherwise} \end{cases} .$$

We want to derive k(r), the pdf of $r = \sqrt{x^2 + y^2}$. Let

$$u = x^2 + y^2$$
$$w = x^2 \qquad \text{(C-1)}$$

or

$$x = \pm\sqrt{w}$$
$$y = \pm\sqrt{u-w} \quad , \qquad \text{(C-2)}$$

pages 92, 93 and 94 Blank

where the sign on the radical depends on the quadrant of A_o (see Figure C.1). The Jacobian J_1 of the transformation (C-2) in quadrant

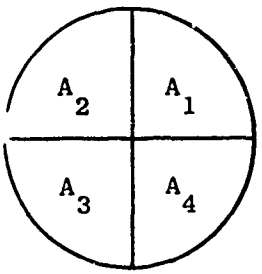


FIGURE C.1 THE FOUR QUADRANTS OF A_o

A_1 is the determinant

$$J_1 = \begin{vmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial w} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial w} \end{vmatrix} = \frac{1}{4} \frac{1}{\sqrt{uw - w^2}} \quad .$$

Similarly, $J_2 = -J_1$, $J_3 = J_1$, and $J_4 = -J_1$, so $|J_i| = |J|$ is the same over all quadrants. The transofrmation (C-1) takes quadrant A_i into a region B_i in the uw- plane. This mapping is illustrated in Figure C.2 and is described by the limits:

$$\left.\begin{array}{ll} x: & 0 \rightarrow R_o \\ y: & 0 \rightarrow R_o \end{array}\right\} \Longrightarrow \left\{\begin{array}{ll} u: & 0 \rightarrow R_o^2 \\ w: & 0 \rightarrow u \end{array}\right. \quad .$$

The quadrants A_2, A_3, A_4, also map onto regions B_2, B_3, and B_4, which are coincident with B_1. Consider B_1 the image of A_1; we can use the theorem on transformation of random variables (e.g., Ref. 10) to write

$$H(u,w) = f(\sqrt{w},\sqrt{u-w})\,|J|$$

$$= \frac{1}{4A_o}\,\frac{1}{\sqrt{uw-w^2}} \qquad \text{for } (u,w)\in B_1 \text{ (image of } A_1) \quad .$$

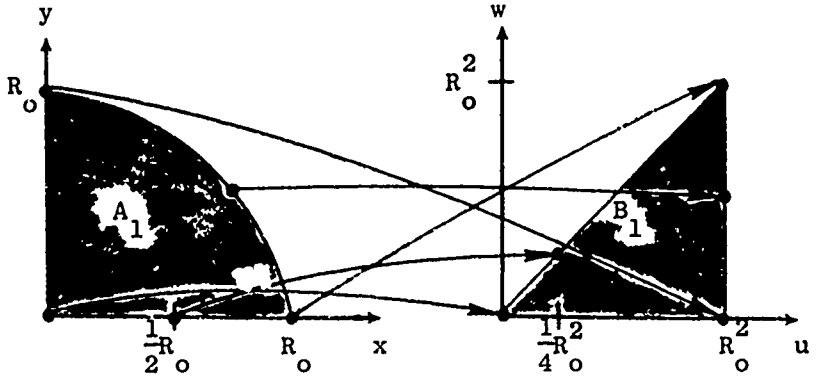


FIGURE C.2 THE MAPPING OF A_1 ONTO B_1

It is clear, however, that B_1 is also the image of A_2, A_3, A_4 so that considering all the transformed probability density we have

$$h(u,w) = \frac{1}{A_o}\,\frac{1}{\sqrt{uw-w^2}} \qquad \text{for } (u,w)\in B_1 \quad .$$

Now we integrate out w in order to obtain the marginal pdf g(u) of u:

$$g(u) = \int_o^u h(u,w)\,dw$$

$$= \frac{\pi}{A_o} \qquad \text{for } 0 \le u \le R_o^2 \quad .$$

Now $r = \sqrt{u}$ so that

$$k(r) = g(r^2)\left|\frac{du}{dr}\right|$$

$$= \frac{2\pi}{A_o} r \qquad \text{for } 0 \le r \le R_o \quad .$$

Appendix D

FORMAC PROGRAM LISTING

This appendix presents a listing of the program written in the IBM FORMAC language to handle the algebraic simplification of the flow graph transmission for the six-state model. A sample output is also given. The listing is presented in order to give the interested reader who is unfamiliar with FORMAC an idea of what the language is like and what it can do. This appendix can only be considered as an accompaniment to Ref. 7 (the FORMAC handbook); the intention is not to provide an introduction to FORMAC but rather an example. Neither can the program be considered efficient; indeed, many improvements are obvious to someone with only limited knowledge of the language. The following correspondences should be noted in reading the program and relating it to the problem:

PROBLEM NOTATION	PROGRAM NOTATION
n	N2, N3
a through f	AA through PP
A through F	A through F
s + a	SPA
p_i (coefficient in P(s))	CUS(i-1)
q_i (coefficient in Q(s))	CDS(i-1)
Δ	DEL, DELTA, DELR
P(s)	UP1
Q(s)	DWN1
$N_Q(n) + 1$	L1

pages 99 and 100 Blank

It is assumed in this listing that n = 1 and that b = c (the simplest case considered--the one given in the main text). It is interesting to note that FORMAC is imbedded in PL/1, so that it is possible to incorporate the numberical computation into the same program as the algebraic manipulation. This was not done on this project because an IBM computation facility was not conveniently accessible, and PL/1-FORMAC is unique to IBM.

```
INPUT TO FORMAC PREPROCESSOR
  PROC2: PROCEDURE OPTIONS(MAIN);
    FORMAC_OPTIONS;
    OPTSET(LINELENGTH=72);
    PUT PAGE;
    PUT LIST('  PROJECT 1318-5 .... FORMAC TRY OUT');
   DCL POWER BIN FIXED(31);
   DCL HPN BIN FIXED(31);
   DCL HPD BIN FIXED(31);
    LET(FNC(FT)= ($(1)**$(2))**$(1)/$(3)**$(1));
    N1=1;
    N2=1;
    N3=1;
    N4=1;
    N5=1;
  BEGIN:
    LET(AX =FT("N1",A,SPA);
        BX =FT("N2",B,SPB);
         CX =FT("N2",B,SPB);
        DX =FT("N4",D,SPD);
        FX =FT("N5",F,SPF) );
    PRINT_OUT(AX;BX;CX;DX;FX) ;
    LET( AA = P12*AX; BB = P13*AX; CC = P23*BX; DD = P26*BX;
         EE = P63*CX; FF = P42*DX; GG = P24*BX; HH = P45*DX;
         KK = P52*FX; MM = P43*DX; NN = P53*FX; PP = P6E*CX;
         QQ = P64*CX );
    PRINT_OUT(AA;BB;CC;DD;EE;FF;GG;HH;KK;MM;NN;PP;QQ;);
    LET( DELTA = 1 - (PP + GG*FF + GG*HH*KK + DD*FF*QQ + DD*HH*KK*QQ)
                 + (FF*GG*PP + GG*HH*KK*PP);
         DEL = CODEM(DELTA) );
    PRINT_OUT (DELTA;DEL);
    ATOMIZE(DELTA);
    ATOMIZE(ABG);
    LET(ABG=DEL);
    ATOMIZE(DEL);
    INDEX=1;
    GO TO ST2;
ST3:
    LET(DEL=ABG);
    ATOMIZE(ABG);
    PRINT_OUT(DEL)
    LET( DELR = REPLACE(DEL,SPA,S+"N1"*A,SPB,S+"N2"*B,
        SPC,S+"N3"*C,SPD,S+"N4"*D,SPF,S+"N5"*F ) );
    ATOMIZE(DELR);
    LET(T2 =     (DD*EE + (CC + GG*MM + GG*HH*NN)*(1-PP)
              + DD*QQ*(MM + HH*NN))/DEL;
        T2R = CODEM(T2) );
    ATOMIZE(DEL);
    LET(ABG=T2R);
    ATOMIZE(T2R);
    INDEX=2;
    GO TO ST2;
 ST4:
    LET(T2R=ABG);
    ATOMIZE(ABG);
    PRINT_OUT(T2R);
    LET( T2   = REPLACE(T2R,SPA,S+"N1"*A,SPB,S+"N2"*B,
        SPC,S+"N3"*C,SPD,S+"N4"*D,SPF,S+"N5"*F ) );
    ATOMIZE(T2R);
    LET(UP1 = EXPAND(NUM(T2 )); DWN = DENOM(T2 );
```

```
          M1 = HIGHPOW(UP1,S));
      LET(UP1 = EXPAND(UP1*S));
      NU = INTEGER(M1);
      DO N=0 TO NU;
      NQ=N+1;
      LET(COS("N")=COEFF(UP1,S**"NQ"));
      PRINT_OUT(COS("N"));
      ATOMIZE(COS("N"));
      END;
      ATOMIZE(UP1);
      LET(DWN1=EXPAND(DWN); L1=HIGHPOW(DWN1,S));
      LET(DWN1 = EXPAND(DWN1*S));
      PRINT_OUT(L1);
      ATOMIZE(DWN);
      ND = INTEGER(L1);
      DO N=0 TO ND;
      NQ=N+1;
      LET(COS("N")=COEFF(DWN1,S**"NQ"));
      PRINT_OUT(COS("N"));
      ATOMIZE(COS("N"));
      END;
      ATOMIZE(DWN1);
      PUT LIST('  END OF REPEAT');
      GO TO ST1;
ST2:
      LET(AT = EXPAND(NUM( ABB ));
          BT = EXPAND(DENOM( ABB ));
          Z(1) = SPA;
          Z(2) = SPB;
          Z(3) = SPC;
          Z(4) = SPD;
          Z(5) = SPE );
      DO I = 1 TO 5;
      ATOMIZE(AS:BS);
      LET(AS=0;BS=0; Y=Z("I"));
      LET(HPNY = HIGHPOW(AT,Y);
          LPNY = LOWPOW(AT,Y);
          HPDY = HIGHPOW(BT,Y);
          LPDY = LOWPOW(BT,Y) );
      HPN = INTEGER(HPNY);
      LPN = INTEGER(LPNY);
      HPD = INTEGER(HPDY);
      LPD = INTEGER(LPDY);
      IF LPN<=0 THEN GO TO ST1;
                                ELSE;
      IF LPD<=0 THEN GO TO ST1;
                                ELSE;
      IF LPN>=LPD THEN POWER = LPD;
                  ELSE POWER = LPN;
      DO K = LPD TO HPD;
      LET(AS = AS + COEFF(AT,Y**"K")*Y**("K" - "POWER") );
      LET(AS=EXPAND(AS));
      END;
      DO K = LPD TO HPD;
      LET(BS = BS + COEFF(BT,Y**"K")*Y**("K" - "POWER") );
      LET(BS=EXPAND(BS));
      END;
      ATOMIZE(AS;BT);
      LET(AT = AS; BT = BS);
ST1:
```

PROGRAM LISTING (concluded)

```
    END;
    LET( ANG  = AT/BT);
    IF INDEX=1 THEN GO TO ST3; ELSE GO TO ST4;
SIE:
    END;
```

SAMPLE OUTPUT

```
 PROJECT 1318-5 .... FORMAC TRY OUT
AX =  A / SPA
-------------
BX =  B / SPB
-------------
CX =  B / SPB
-------------
DX =  D / SPD
-------------
FX =  F / SPF
-------------
AA = P12 A /  SPA
----------------
BB = P13 A /  SPA
----------------
CC = P23 B /  SPB
----------------
DD = P26 B /  SPB
----------------
FF = P63 B /  SPB
----------------
FF =  P42 D / SPD
----------------
GG = P24 B /  SPB
----------------
HH =  P45 D / SPD
----------------
KK =  F P52 / SPF
----------------
MM =  D P43 / SPD
----------------
NN =  F P53 / SPF
----------------
PP =  B P66 / SPB
----------------
QQ =  B P64 / SPB
----------------
                            2                               2                   2
DELTA =  F P45 P24 D B   P66 P52 / ( SPF SPD SPB   ) -  F F45 P26 D B
--------------------------------------------------------------------------------
                         2
P64 P52 / ( SPF SPD SPB    ) -  F P45 P24 D B F52 / ( SPF SPD SPB ) -  B
--------------------------------------------------------------------------------
                         2                 2                     2
P66 / SPB +  P24 P42 D B   P66 / ( SPD SPB   ) -  P42 P26 D B   P64 / (
--------------------------------------------------------------------------------
       2
SPD SPB   ) -  P24 P42 D B / ( SPD SPB ) + 1
--------------------------------------------------------
              3     5                  10                  3     4
DEL = (  - SPF   SPD   P24 P42 D SPB    B + (  - SPF   SPD   P42 P26 D
--------------------------------------------------------------------------------
    4   2              3     4                  4   2                3      3
SPB   B   P64 + ( SPF   SPD   P24 P42 D SPB   B   P66 + (  - SPF   SP
--------------------------------------------------------------------------------
     5                  2       2                 4
 SPB   B P66 + (  - SPF   F SPD   P45 P24 D SPB   B P52 + ( SPF F SPD
--------------------------------------------------------------------------------
                 4  2                                     2  2
P45 P24 D SPB   B   P66 P52 - SPF F SPD P45 P26 D SPB   B   P64 P52 )
```

SAMPLE OUTPUT (continued)

```
---------------------------------------------------------------------
                            2            2                  3     6     11
SPF  SPD SPB ) SPB ) SPD SPB    ) SPD SPB    ) SPD SPB + SPF    SPD    SPB
---------------------------------------------------------------------
           3      6      11
 ) / (  SPF    SPD    SPB      )
---------------------------
                       2                     2
DEL  = ( F P45 P24 D B    P66 P52 - F P45 P26 D B    P64 P52 - F P45 P24 D
---------------------------------------------------------------------
                                             2                          2
SPB B P52 - SPF SPD SPB B P66 + SPF P24 P42 D B    P64 - SPF P42 P26 D B
---------------------------------------------------------------------
                                    2                       2
 P64 - SPF P24 P42 D SPB B + SPF SPD SPB    ) / (  SPF SPD SPB    )
--------------------------------------------------------------
                       2                     2
T2R  =  (  - SPF P24 D B    P66 P43 + SPF P26 D B    P64 P43 + SPF P24 D
---------------------------------------------------------------------
                         2                    2
SPB B P43 - F P45 P24 D B    P66 P53 + F P45 P26 D B    P64 P53 + F P45
---------------------------------------------------------------------
                            2
P24 D SPB B P53 - P23 SPF SPD B    P66 + P23 SPF SPD SPB B + SPF SPD P63
---------------------------------------------------------------------
      2                      2                     2
P26 B    ) / ( F P45 P24 D B    P66 P52 - F P45 P26 D B    P64 P52 - F P45
---------------------------------------------------------------------
                                                     2
P24 D SPB B P52 - SPF SPD SPB B P66 + SPF P24 P42 D B    P66 - SPF P42
---------------------------------------------------------------------
        2                                          2
P26 D B    P64 - SPF P24 P42 D SPB B + SPF SPD SPB    )
--------------------------------------------------
                         2                   2                     2
CUS(0)  =   - F P24 D B    P66 P43 + F P26 D B    P64 P43 + F P24 D B    P43
---------------------------------------------------------------------
                  2                      2                          2
 - F P45 P24 D B    P66 P53 + F P45 P26 D B    P64 P53 + F P45 P24 D B
---------------------------------------------------------------------
                   2                 2              2
P53 - P23 F D B    P66 + F P63 P26 D B    + P23 F D B
----------------------------------------------
                      2                 2
CUS(1)  =   - P24 D B    P66 P43 + P26 D B    P64 P43 + F P24 D B P43 + P24
---------------------------------------------------------------------
    2                                       2              2
D B    P43 + F P45 P24 D B P53 - P23 D B    P66 - P23 F B    P66 + P23 F D
---------------------------------------------------------------------
                   2           2                2            2
B + P63 P26 D B    + P23 D B    + F P63 P26 B    + P23 F B
--------------------------------------------------------
                                   2                                      2
CUS(2) = P24 D B P43 - P23 B    P66 + P23 D B + P23 F B + P63 P26 B    +
---------------------------------------------------------------------
      2
P23 B
-------
CUS(3) = P23 B
--------------
```

```
L1 = 4
------

                                  2                                  2
COS(0) = F P45 P24 L B    P66 P52 - F P45 P26 D B    P64 P52 - F P45 P24 D
-------------------------------------------------------------------------

  2                                 2                 2                 2
 B    P52 + F P24 P42 D B    P66 - F D B    P66 - F P42 P26 L B    P64 - F
-------------------------------------------------------------------------

            2           2
P24 P42 D B    + F D B
-----------------------

                                                  2                  2
COS(1) =  - F P45 P24 D B P52 - F D B P66 - D F    P66 + P24 P42 D F
-------------------------------------------------------------------------

          2                 2                                  2
P66 - F B    P66 - P42 P26 D B    P64 - F P24 P42 D B + 2 F D B + D F    -
-------------------------------------------------------------------------

            2           2
P24 P42 L B    + F B
--------------------

                                  2
COS(2) =  - D B P66 - F B P66 - B    P66 - P24 P42 D B + 2 D B + 2 F B +
-------------------------------------------------------------------------

       2
F D + B
---------
COS(3) = 2 B + D + F - B P66
---------------------------
COS(4) = 1
----------
  END OF REPORT
```

Appendix E

POLYNOMIAL COEFFICIENTS

This appendix contains the coefficients of the polynomials P(s) and Q(s) for n = 2, 3, and 10. The coefficients are presented in FORTRAN format due to the fact that they have been printed directly from verified punched cards in order to eliminate typographical errors. To assist interpretation of the coefficients the following correspondences between model terminology and FORTRAN are noted:

Model	FORTRAN
' (multiplication)	*
a,b,c,d,f	A,B,C,D,F
a^2, b^2, c^2, d^2, f^2	A2,B2,C2,D2,F2
p_{ij} (e.g., p_{26})	PIJ (e.g., P26)
p_i	P(i)
q_i	Q(i)

The P(i) and Q(i) are given in Tables E.1 through E.3.

pages 109 and 110 Blank

Table E.1

COEFFICIENTS FOR n = 2

```
P(1) =  16*F*D*P26*C2*B2*P53*P45*P64
      +16*F*D*P26*C2*B2*P43*P64
      -16*F*D*P24*C2*B2*P66*P53*P45
      +16*F*D*P24*C2*B2*P53*P45
      -16*F*D*P24*C2*B2*P66*P43
      +16*F*D*P24*C2*B2*P43
      -16*F*D*P23*C2*B2*P66
      +16*F*D*P23*C2*B2
      +16*F*D*P63*P26*C2*B2

P(2) =  16*D*P26*C2*B2*P43*P64
      +16*F*D*P24*C*B2*P53*P45
      -16*D*P24*C2*B2*P66*P43
      +16*F*D*P24*C*B2*P43
      +16*D*P24*C2*B2*P43
      -16*D*P23*C2*B2*P66
      -16*F*P23*C2*B2*P66
      +16*F*D*P23*C*B2
      +16*D*P23*C2*B2
      +16*F*P23*C2*B2
      +16*D*P63*P26*C2*B2
      +16*F*P63*P26*C2*B2

P(3) =  4*F*D*P24*B2*P53*P45
      +16*D*P24*C*B2*P43
      +4*F*D*P24*B2*P43
      -16*P23*C2*B2*P66
      +16*D*P23*C*B2
      +16*F*P23*C*B2
      +4*F*D*P23*B2
      +16*P23*C2*B2
      +16*P63*P26*C2*B2

P(4) =  4*D*P24*B2*P43
      +16*P23*C*B2
      +4*D*P23*B2
      +4*F*P23*B2

P(5) =  4*P23*B2

Q(1) = -16*F*D*P26*C2*B2*P52*P45*P64
      -16*F*D*P24*C2*B2*P26*C2*B2*P64
      +16*F*D*P24*C2*B2*P66*P52*P45
      -16*F*D*P24*C2*B2*P52*P45
      +16*F*D*P24*P42*C2*B2*P66
      -16*F*D*C2*B2*P66
      -16*F*D*P24*P42*C2*B2
      +16*F*D*C2*B2
```

Table E.1 (continued)

```
Q(2) = -16*D*P42*P26*C2*B2*P64
       -16*F*D*P24*C*B2*P52*P45
       -16*F*D*C2*B*P66
       +16*D*P24*P42*C2*B2*P66
       -16*D*C2*B2*P66
       -16*F*C2*B2*P66
       +16*F*D*C2*B
       -16*F*D*P24*P42*C*B2
       +16*F*D*C*B2
       -16*D*P24*P42*C2*B2
       +16*D*C2*B2
       +16*F*C2*B2

Q(3) = -4*F*D*P24*B2*P52*P45
       -16*D*C2*B*P66
       -16*F*C2*B*P66
       -16*C2*B2*P66
       -4*F*D*C2*P66
       +16*F*D*C*B
       +16*D*C2*B
       +16*F*C2*B
       -16*D*P24*P42*C*B2
       +16*D*C*B2
       +16*F*C*B2
       -4*F*D*P24*P42*B2
       +4*F*D*B2
       +16*C2*B2
       -4*F*D*C2

Q(4) = -16*C2*B*P66
       -4*D*C2*P66
       -4*F*C2*P66
       +16*D*C*B
       +16*F*C*B
       +4*F*D*B
       +16*C2*B
       +4*F*D*C
       +16*C*B2
       -4*D*P24*P42*B2
       +4*D*B2
       +4*F*B2
       +4*D*C2
       +4*F*C2
```

Table E.1 (concluded)

```
Q(5) = -4*C2*P66
       +16*C*B
       +4*D*B
       +4*F*B
       +4*D*C
       +4*F*C
       +F*D
       +4*B2
       +4*C2

Q(6) =  4*B
       +4*C
       +D
       +F

Q(7) =  1
```

Table E.2

COEFFICIENTS FOR n = 3

```
P(1) =  729*F*D*P26*C3*B3*P53*P45*P64
      +729*F*D*P26*C3*B3*P43*P64
      -729*F*D*P24*C3*B3*P66*P53*P45
      +729*F*D*P24*C3*B3*P53*P45
      -729*F*D*P24*C3*B3*P66*P43
      +729*F*D*P24*C3*B3*P43
      -729*F*D*P23*C3*B3*P66
      +729*F*D*P23*C3*B3
      +729*F*D*P63*P26*C3*B3

P(2) =  729*D*P26*C3*B3*P43*P64
      +729*F*D*P24*C2*B3*P53*P45
      -729*D*P24*C3*B3*P66*P43
      +729*F*D*P24*C2*B3*P43
      +729*D*P24*C3*B3*P43
      -729*D*P23*C3*B3*P66
      -729*F*P23*C3*B3*P66
      +729*F*D*P23*C2*B3
      +729*D*P23*C3*B3
      +729*F*P23*C3*B3
      +729*D*P63*P26*C3*B3
      +729*F*P63*P26*C3*B3

P(3) =  243*F*D*P24*C*B3*P53*P45
      +243*F*D*P24*C*B3*P43
      +729*D*P24*C2*B3*P43
      -729*P23*C3*B3*P66
      +243*F*D*P23*C*B3
      +729*D*P23*C2*B3
      +729*F*P23*C2*B3
      +729*P23*C3*B3
      +729*P63*P26*C3*B3

P(4) =  27*F*D*P24*B3*P53*P45
      +243*D*P24*C*B3*P43
      +27*F*D*P24*B3*P43
      +243*D*P23*C*B3
      +243*F*P23*C*B3
      +27*F*D*P23*B3
      +729*P23*C2*B3

P(5) =  27*D*P24*B3*P43
      +243*P23*C*B3
      +27*D*P23*B3
      +27*F*P23*B3

P(6) =  27*P23*B3
```

Table E.1 (continued)

```
Q(1) = -729*F*D*P26*C3*B3*P52*P45*P64
       -729*F*D*P42*P26*C3*B3*P64
       +729*F*D*P24*C3*B3*P66*P52*P45
       -729*F*D*P24*C3*B3*P52*P45
       +729*F*D*P24*P42*C3*B3*P66
       -729*F*D*C3*B3*P66
       -729*F*D*P24*P42*C3*B3
       +729*F*D*C3*B3

Q(2) = -729*D*P42*P26*C3*B3*P64
       -729*F*D*P24*C2*B3*P52*P45
       -729*F*D*C3*B2*P66
       +729*D*P24*P42*C3*B3*P66
       -729*D*C3*B3*P66
       -729*F*C3*B3*P66
       +729*F*D*C3*B2
       -729*F*D*P24*P42*C2*B3
       +729*F*D*C2*B3
       -729*D*P24*P42*C3*B3
       +729*D*C3*B3
       +729*F*C3*B3

Q(3) = -243*F*D*P24*C*B3*P52*P45
       -243*F*D*C3*B*P66
       -729*D*C3*B2*P66
       -729*F*C3*B2*P66
       -729*C3*B3*P66
       +243*F*D*C3*B
       +729*F*D*C2*B2
       +729*D*C3*B2
       +729*F*C3*B2
       -243*F*D*P24*P42*C*B3
       +243*F*D*C*B3
       -729*D*P24*P42*C2*B3
       +729*D*C2*B3
       +729*F*C2*B3
       +729*C3*B3
```

Table E.2 (continued)

```
Q(4) = -27*F*D*P24*B3*P52*P45
       -243*D*C3*B*P66
       -243*F*C3*B*P66
       -729*C3*B2*P66
       -27*F*D*C3*P66
       +243*F*D*C2*B
       +243*D*C3*B
       +243*F*C3*B
       +243*F*D*C*B2
       +729*D*C2*B2
       +729*F*C2*B2
       +729*C3*B2
       -243*D*P24*P42*C*B3
       +243*D*C*B3
       +243*F*C*B3
       -27*F*D*P24*P42*B3
       +27*F*D*B3
       +729*C2*B3
       +27*F*D*C3

Q(5) = -243*C3*B*P66
       -27*D*C3*P66
       -27*F*C3*P66
       +81*F*D*C*B
       +243*D*C2*B
       +243*F*C2*B
       +243*C3*B
       +243*D*C*B2
       +243*F*C*B2
       +27*F*D*B2
       +729*C2*B2
       +243*C*B3
       -27*D*P24*P42*B3
       +27*D*B3
       +27*F*B3
       +27*F*D*C2
       +27*D*C3
       +27*F*C3
```

Table E.2 (concluded)

```
Q(6) = -27*C3*P66
       +81*D*C*B
       +81*F*C*B
       +9*F*D*B
       +243*C2*B
       +9*F*D*C
       +243*C*B2
       +27*D*B2
       +27*F*B2
       +27*D*C2
       +27*F*C2
       +27*B3
       +27*C3

Q(7) =  81*C*B
       +9*D*B
       +9*F*B
       +9*D*C
       +9*F*C
       +F*D
       +27*B2
       +27*C2

Q(8) =  9*B
       +9*C
       +D
       +F

Q(9) =  1
```

Table E.2 (concluded)

```
Q(6) = -27*C3*P66
       +81*D*C*B
       +81*F*C*B
       +9*F*D*B
       +243*C2*B
       +9*F*D*C
       +243*C*B2
       +27*D*B2
       +27*F*B2
       +27*D*C2
       +27*F*C2
       +27*B3
       +27*C3

Q(7) =  81*C*B
       +9*D*B
       +9*F*B
       +9*D*C
       +9*F*C
       +F*D
       +27*B2
       +27*C2

Q(8) =  9*B
       +9*C
       +D
       +F

Q(9) =  1
```

Table E.3

COEFFICIENTS FOR n = 10

```
P(1)  = (-F*P24*D*B20*P66*P43
        +F*P26*D*B20*P64*P43
        +F*P24*D*B20*P43
        -F*P45*P24*D*B20*P66*P53
        +F*P45*P26*D*B20*P64*P53
        +F*P45*P24*D*B20*P53
        -P23*F*D*B20*P66
        +F*P63*P26*D*B20
        +P63*F*D*B20)*1E+20

P(2)  = (-P24*D*B20*P66*P43
        +P26*D*B20*P64*P43
        +F*P24*D*B19*P43
        +P24*D*B20*P43
        +F*P45*P24*D*B19*P53
        -P23*D*B20*P66
        -P23*F*B20*P66
        +P23*F*D*B19
        +P63*P26*D*B20
        +P23*D*B20
        +F*P63*P26*B20
        +P23*F*B20)*1E+20

P(3)  = (.45*F*P24*D*B18*P43
        +P24*D*B19*P43
        +.45*F*P45*P24*D*B18*P53
        -P23*B20*P66
        +.45*P23*F*D*B18
        +P23*D*B19
        +P23*F*B19
        +P63*P26*B20
        +P23*B20)*1E+20

P(4)  = (.12*F*P24*D*B17*P43
        +.45*P24*D*B18*P43
        +.12*F*P45*P24*D*B17*P53
        +.12*P23*F*D*B17
        +.45*P23*D*B18
        +.45*P23*F*B18
        +P23*B19)*1E+20

P(5)  = (.21*F*P24*D*B16*P43
        +1.2*P24*D*B17*P43
        +.21*F*P45*P24*D*B16*P53
        +.21*P23*F*D*B16
        +1.2*P23*D*B17
        +1.2*P23*F*B17
        +4.5*P23*B18)*1E+19
```

Table E.3 (continued)

```
P(6)  = (.252*F*P24*D*B15*P43
       +2.1*P24*D*B16*P43
       +.252*F*P45*P24*D*B15*P53
       +.252*P23*F*D*B15
       +2.1*P23*D*B16
       +2.1*P23*F*B16
       +12.*P23*B17)*1E+18

P(7)  = (.21*F*P24*D*B14*P43
       +2.52*P24*D*B15*P43
       +.21*F*P45*P24*D*B14*P53
       +.21*P23*F*D*B14
       +2.52*P23*D*B15
       +2.52*P23*F*B15
       +21.*P23*B16)*1E+17

P(8)  = (.12*F*P24*D*B13*P43
       +2.1*P24*D*B14*P43
       +.12*F*P45*P24*D*B13*P53
       +.12*P23*F*D*B13
       +2.1*P23*D*B14
       +2.1*P23*F*B14
       +25.2*P23*B15)*1E+16

P(9)  = (.45*F*P24*D*B12*P43
       +12.*P24*D*B13*P43
       +.45*F*P45*P24*D*B12*P53
       +.45*P23*F*D*B12
       +12.*P23*D*B13
       +12.*P23*F*B13
       +210.*P23*B14)*1E+14

P(10) = (F*P24*D*B11*P43
       +45.*P24*D*B12*P43
       +F*P45*P24*D*B11*P53
       +P23*F*D*B11
       +45.*P23*D*B12
       +45.*P23*F*B12
       +1200.*P23*B13)*1E+12

P(11) = (F*P24*D*B10*P43
       +100.*P24*D*B11*P43
       +F*P45*P24*D*B10*P53
       +P23*F*D*B10
       +100.*P23*D*B11
       +100.*P23*F*B11
       +4500.*P23*B12)*1E+10
```

Table E.3 (continued)

```
P(12) = (P24*D*B10*P43
        +P23*D*B10
        +P23*F*B10
        +100.*P23*B11)*1E+10

P(13) =  P23*B10*1E+10

Q(1)  = (F*P45*P24*D*B20*P66*P52
        -F*P45*P26*D*B20*P64*P52
        -F*P45*P24*D*B20*P52
        +F*P24*P42*D*B20*P66
        -F*D*B20*P66
        -F*P42*P26*D*B20*P64
        -F*P24*P42*D*B20
        +F*D*B20)*1E+20

Q(2)  = (-F*P45*P24*D*B19*P52
        -F*D*B19*P66
        +P24*P42*D*B20*P66
        -D*B20*P66
        -F*B20*P66
        -P42*P26*D*B20*P64
        -F*P24*P42*D*B19
        +2.*F*D*B19
        -P24*P42*D*B20
        +D*B20
        +F*B20)*1E+20

Q(3)  = (-.45*F*P45*P24*D*B18*P52
        -.45*F*D*B18*P66
        -D*B19*P66
        -F*B19*P66
        -B20*P66
        -.45*F*P24*P42*D*B18
        +1.9*F*D*B18
        -P24*P42*D*B19
        +2.*D*B19
        +2.*F*B19
        +B20)*1E+20
```

Table E.3 (continued)

```
Q(4)  = (-.12*F*P45*P24*D*B17*P52
         -.12*F*D*B17*P66
         -.45*D*B18*P66
         -.45*F*B18*P66
         -B19*P66
         -.12*F*P24*P42*D*B17
         +1.14*F*D*B17
         -.45*P24*P42*D*B18
         +1.9*D*B18
         +1.9*F*B18
         +2.*B19)*1E+20

Q(5)  = (-.21*F*P45*P24*D*B16*P52
         -.21*F*D*B16*P66
         -1.2*D*B17*P66
         -1.2*F*B17*P66
         -4.5*B18*P66
         -.21*F*P24*P42*D*B16
         +4.845*F*D*B16
         -1.2*P24*P42*D*B17
         +11.4*D*B17
         +11.4*F*B17
         +19.*B18)*1E+19

Q(6)  = (-.252*F*P45*P24*D*B15*P52
         -.252*F*D*B15*P66
         -2.1*D*B16*P66
         -2.1*F*B16*P66
         -12.*B17*P66
         -.252*F*P24*P42*D*B15
         +15.504*F*D*B15
         -2.1*P24*P42*D*B16
         +48.45*D*B16
         +48.45*F*B16
         +114.*B17)*1E+18

Q(7)  = (-.21*F*P45*P24*D*B14*P52
         -.21*F*D*B14*P66
         -2.52*D*B15*P66
         -2.52*F*B15*P66
         -21.*B16*P66
         -.21*F*P24*P42*D*B14
         +38.76*F*D*B14
         -2.52*P24*P42*D*B15
         +155.04*D*B15
         +155.04*F*B15
         +484.5*B16)*1E+17
```

Table E.3 (continued)

```
Q(8)  = (-.12*F*P45*P24*D*B13*P52
         -.12*F*D*B13*P66
         -2.1*D*B14*P66
         -2.1*F*B14*P66
         -25.2*B15*P66
         -.12*F*P24*P42*D*B13
         +77.52*F*D*B13
         -2.1*P24*P42*D*B14
         +387.6*D*B14
         +387.6*F*B14
         +1550.4*B15)*1E+16

Q(9)  = (-.45*F*P45*P24*D*B12*P52
         -.45*F*D*B12*P66
         -12.*D*B13*P66
         -12.*F*B13*P66
         -210.*B14*P66
         -.45*F*P24*P42*D*B12
         +1259.7*F*D*B12
         -12.*P24*P42*D*B13
         +7752.*D*B13
         +7752.*F*B13
         +38760.*B14)*1E+14

Q(10) = (-F*P45*P24*D*B11*P52
         -F*D*B11*P66
         -45.*D*B12*P66
         -45.*F*B12*P66
         -1200.*B13*P66
         -F*P24*P42*D*B11
         +16796.*F*D*B11
         -45.*P24*P42*D*B12
         +125970.*D*B12
         +125970.*F*B12
         +775200.*B13)*1E+12

Q(11) = (-F*P45*P24*D*B10*P52
         -F*D*B10*P66
         -100.*D*B11*P66
         -100.*F*B11*P66
         -4500.*B12*P66
         -F*P24*P42*D*B10
         +16756.*F*D*B10
         -100.*P24*P42*D*B11
         +1679600.*D*B11
         +1679600.*F*B11
         +12597000.*B12)*1E+10
```

Table E.3 (continued)

```
Q(12) = (-D*B10*P66
         -F*B10*P66
         -100.*B11*P66
         +16796.*F*D*B9
         -P24*P42*D*B10
         +184756.*D*B10
         +184756.*F*B10
         +1679600.*B11)*1E+10

Q(13) = (-B10*P66
         +1259.7*F*D*B8
         +16796.*D*B9
         +16796.*F*B9
         +184756.*B10)*1E+10

Q(14) = (.7752*F*D*B7
         +12.597*D*B8
         +12.597*F*B8
         +167.96*B9)*1E+12

Q(15) = (.3876*F*D*B6
         +7.752*D*B7
         +7.752*F*B7
         +125.97*B8)*1E+11

Q(16) = (15.504*F*D*B5
         +387.6*D*B6
         +387.6*F*B6
         +7752.*B7)*1E+8

Q(17) = (4.845*F*D*B4
         +155.04*D*B5
         +155.04*F*B5
         +3876.*B6)*1E+7

Q(18) = (114.*F*D*B3
         +4845.*D*B4
         +4845.*F*B4
         +155040.*B5)*1E+4

Q(19) =  19000.*F*D*B2
         +1140000.*D*B3
         +1140000.*F*B3
         +48450000.*B4
```

Table E.3 (concluded)

```
Q(20) =  200.*F*D*B
              +19000.*D*B2
              +19000.*F*B2
              +1140000.*B3

Q(21) =  200.*D*B
              +200.*F*B
              +F*D
              +19000.*B2

Q(22) =  200.*B
              +D
              +F

Q(23) =  1.
```